

**VIỆN KHOA HỌC VÀ CÔNG NGHỆ VIỆT NAM
VIỆN CÔNG NGHỆ THÔNG TIN**

BÁO CÁO TỔNG KẾT ĐỀ TÀI NGHỊ ĐỊNH THƯ

**HỢP TÁC NGHIÊN CỨU PHÁT TRIỂN
CÁC HỆ THỐNG XỬ LÝ ẢNH NHANH TRÊN CƠ SỞ
ÁP DỤNG CÔNG NGHỆ MẠNG NƠON
PHI TUYẾN TẾ BÀO**

Chủ nhiệm đề tài: PGS. TSKH. PHẠM THƯỢNG CÁT

6730

19/02/2008

HÀ NỘI - 2007

MỤC LỤC

	<i>Trang</i>
1. BÁO CÁO KẾT QUẢ KHẢO SÁT, NGHIÊN CỨU MẠNG NƠON TẾ BÀO VÀ CÔNG NGHỆ XỬ LÝ ẢNH NHANH TRÊN MẠNG NƠON TẾ BÀO CNN	01
1.1. Mở đầu	01
1.2. Mạng noron tế bào CNN	03
1.3. Máy tính vạn năng mạng noron tế bào CNN – UM	32
1.4. Công nghệ xử lý ảnh nhanh trên nền mạng CNN	39
1.4.1. Máy tính xử lý ảnh nhanh CNN Bi – I	39
1.4.2. Hệ phần mềm phát triển Bi – I	46
1.4.3. Thư viện xử lý ảnh InstantVision	55
1.5. Một số phương pháp xử lý theo công nghệ mạng CNN	71
1.5.1. Thiết kế các mẫu (A, B, z) cho mạng CNN	71
1.5.2. Mô hình hóa phương trình đạo hàm riêng sử dụng mạng CNN	81
1.5.3. Mô hình mắt nhân tạo sử dụng mạng CNN	86
1.5.4. Phương pháp xử lý ảnh vân tay sử dụng mạng CNN	91
1.6. Khả năng ứng dụng của CNN	97
1.6.1. Khả năng ứng dụng công nghệ CNN trong công nghiệp và các ngành kinh tế	97
1.6.2. Nhu cầu và tiềm năng ứng dụng công nghệ CNN cho quốc phòng và an ninh	100
1.7. Một số kết quả chính về nghiên cứu phát triển công nghệ CNN tại Viện MTA SzTAKI Hungary thời gian gần đây	104
2. XÂY DỰNG CÁC MÔ HÌNH VÀ THỬ NGHIỆM CÔNG NGHỆ XỬ LÝ ẢNH NHANH CNN PHỤC VỤ CHO NGHIÊN CỨU VÀ ĐÀO TẠO	107
2.1. Mô hình phát tia lửa điện phục vụ cho thí nghiệm thu ảnh tốc độ cao	107
2.2. Mô hình nhận dạng kiểm tra sản phẩm tốc độ cao phục vụ cho nghiên cứu và đào tạo	120
2.3. Thí nghiệm kiểm tra nhanh đai ốc đường sắt sử dụng công nghệ CNN	147
2.4. Thử nghiệm khả năng thu ảnh nhanh các sự kiện thay đổi đột ngột bằng thí nghiệm nổ bong bóng	153
3. KIẾN NGHỊ PHƯƠNG HƯỚNG PHÁT TRIỂN VÀ ỨNG DỤNG CÔNG NGHỆ Ở VIỆT NAM	165
4. CÁC ÁN PHẨM ĐÃ CÔNG BỐ	169
TÀI LIỆU THAM KHẢO	170

1. BÁO CÁO KẾT QUẢ KHẢO SÁT, NGHIÊN CỨU MẠNG NƠN TẾ BÀO VÀ CÔNG NGHỆ XỬ LÝ ẢNH TỐC ĐỘ CAO TRÊN CƠ SỞ MẠNG NƠN TẾ BÀO

Mạng nơ ron tế bào và công nghệ xử lý ảnh tốc độ cao trên cơ sở mạng nơ ron tế bào là một lĩnh vực khoa học công nghệ mới ở Việt nam và trên thế giới; có nhiều triển vọng cho nhiều ứng dụng đột phá. Mục tiêu của nhiệm vụ hợp tác qua đường nghị định thư với Hungary là tiếp nhận và làm chủ được công nghệ xử lý ảnh nhanh, xử lý song song trên nền mạng nơ ron tế bào. Phần báo cáo này giới thiệu tóm tắt các kết quả nghiên cứu tiếp cận công nghệ mới mẻ này đã đạt được của nhiệm vụ.

Mở đầu

Công nghệ xử lý trên cơ sở mạng nơ ron tế bào CNN (Cellular Neural Networks) đã được các nhà khoa học Mỹ và Hungary phát minh vào năm 1992 có tốc độ xử lý 10^{12} phép tính/giây và được áp dụng cho các hệ thống xử lý ảnh nhanh 10-50000 ảnh/giây. Đây là bước đột phá về chất do cấu trúc của máy tính xử lý CNN là song song với hàng chục ngàn CPU được kết nối thành mạng nơ ron trong một chip. Công nghệ này cho phép giải quyết nhiều bài toán xử lý phức tạp trong thời gian thực mà các máy tính thông thường chưa làm được. Do là một phát minh mới trên nền tảng mạng nơ ron, xử lý song song nên hàng loạt các hội nghị quốc tế về CNN đã được tổ chức trên thế giới thời gian qua và đã tạo nền tảng khoa học cho công nghệ CNN. Với tốc độ xử lý 10^{12} phép tính/giây và xử lý ảnh 10-50000 ảnh/giây ranh giới giữa xử lý tín hiệu tương tự và số không còn nhiều khác biệt.

Các máy tính số sử dụng các bộ vi xử lý với hệ lệnh nối tiếp đã phát triển mạnh trong vài chục năm nay. Mặc dù có các nỗ lực trong việc cải tiến nguyên lý hoạt động của các bộ vi xử lý như xử lý ống lệnh (pipeline), siêu luồng (hyper threading) cùng với việc tăng tốc độ xung đồng hồ làm việc của chip vi xử lý, nhưng tuy vậy về cơ bản vẫn là các bộ xử lý với các hệ lệnh nối tiếp. Việc giải các phương trình sóng phụ thuộc không gian thời gian nhanh trong khoảng thời gian rất ngắn (chẳng hạn một vài micro giây) vẫn còn là thách thức với các máy tính tính toán hiện hành.

Trong nhiều lĩnh vực, yêu cầu về các máy tính có công suất tính toán cực mạnh là rất cấp thiết, như trong xử lý ảnh động thời gian thực, nhận dạng và định vị đa mục tiêu di động trong an ninh quốc phòng, kiểm tra chất lượng sản phẩm chuyên động nhanh trên dây chuyền công nghiệp, xử lý chất lượng ảnh siêu âm trong y tế, chế tạo robot thông minh, chế tạo các thiết bị không người lái...

Gần đây phần lớn những nhà sản xuất bộ vi xử lý trên thế giới đã nhận thấy một trong những thách thức lớn cho công nghệ thông tin trong thời gian sắp tới là tạo được một bộ xử lý có hiệu suất cao và một công nghệ nền để có thể biểu diễn được hình ảnh và video trong thời gian thực hoặc xử lý những tín hiệu ở cùng một thời điểm nhưng thu được từ những nguồn khác nhau trong không gian. Cả hai nhiệm vụ này đều liên quan đến tính toán không gian-thời gian. Việc sử dụng phương trình vi phân đạo hàm riêng rời rạc phi tuyến (Nonlinear Partial Difference Equation-PDE) có thể giúp cho máy tính thực hiện được những tính toán này đã có một ảnh hưởng rất lớn. Khả năng lợi dụng những tiềm năng tính toán tương tự theo mảng tín hiệu thay cho cách tính toán số truyền thống theo dòng bit được đề cập đến như một giải pháp mới. Mô hình mạng nơ ron tế bào hay phi tuyến tế bào CNN (Cellular Neural/Nonlinear Network) đã thể hiện đầy đủ khái niệm, giới thiệu một mô hình tính toán mới cho quá trình xử lý ma trận hỗn hợp tín hiệu tương tự và logic. Từ khía cạnh xử lý siêu đẳng kết hợp với khả năng lập trình của CNN

đã đưa tới khái niệm máy tính vạn năng tương tự-logic dựa trên mạng nơ ron tế bào (Cellular Neural Network Universal Machine - CNN-UM). Các CNN-UM thế hệ đầu đã tỏ rõ những ưu thế mà chưa bộ xử lý số nào đáp ứng được. Các máy tính CNN-UM trong những thế hệ sau được phát triển theo hướng mở rộng cấu trúc với đặc tính học (learning) và tự thích nghi (adaptive) sẽ cho chúng ta các máy tính tương tự-logic siêu mạnh và thông minh đủ đáp ứng nhiều đòi hỏi khắt khe về tính toán và xử lý trong thực tiễn.

Lĩnh vực xử lý ảnh số tĩnh và xử lý ảnh động (video) đã được hình thành và phát triển vào những thập kỷ đầu của thế kỷ XX. Các phương pháp xử lý ảnh bắt nguồn từ một số ứng dụng như nâng cao chất lượng thông tin hình ảnh đối với mắt người và xử lý số liệu, nhận dạng cho hệ thống tự động. Một trong những ứng dụng đầu tiên của xử lý ảnh là nâng cao chất lượng ảnh báo truyền qua cáp giữa London và New York vào những năm 1920. Thiết bị đặc biệt mã hóa hình ảnh (báo), truyền qua cáp và khôi phục lại ở phía thu. Cùng với thời gian, do kỹ thuật máy tính phát triển nên xử lý hình ảnh ngày càng phát triển. Các kỹ thuật cơ bản cho phép nâng cao chất lượng hình ảnh như làm nổi đường biên và lưu hình ảnh.

Từ năm 1964 đến nay, phạm vi xử lý ảnh và video (ảnh động) phát triển không ngừng. Các kỹ thuật xử lý ảnh số (digital image processing) đang được sử dụng để giải quyết một loạt các vấn đề nhằm nâng cao chất lượng thông tin hình ảnh. Và xử lý ảnh số được ứng dụng rất nhiều trong y tế, thiên văn học, viễn thám, sinh học, y tế hạt nhân, quân sự, sản xuất công nghiệp... Một ứng dụng rất quan trọng của xử lý ảnh số mà ta không thể không nhắc đến, đó là ứng dụng xử lý ảnh trong lĩnh vực thị giác máy gắn liền với cảm nhận của máy móc tự động. Trong đó, quá trình xử lý thông tin hình ảnh và trích ra những thông tin cần thiết cho bài toán nhận dạng ảnh được sử dụng khá nhiều trong thực tế. Một số vấn đề điển hình ứng dụng kỹ thuật xử lý ảnh tĩnh và ảnh động như tự động nhận dạng chữ in và chữ viết tay, nhận dạng và bám mục tiêu trong quân sự, thị giác máy trong công nghiệp để giám sát, điều khiển và kiểm tra sản phẩm trong dây chuyền sản xuất, tự động nhận dạng vân tay...

Mạng nơ ron tế bào (Cellular Neural Networks – CNN) là một hệ xử lý song song có rất nhiều ứng dụng và khái niệm mới trong nhiều lĩnh vực. Chíp nơ ron tế bào đã thúc đẩy sự ra đời của các thế hệ máy tính xử lý ảnh có tốc độ xử lý cực nhanh. Một trong số chúng là máy tính Bi-I của hãng Analogic Computer Ltd sử dụng chip CNN ACE16k có độ phân giải 128x128 pixel. Máy tính này còn được tích hợp một bộ xử lý tín hiệu số chất lượng cao DSP cung cấp dữ liệu cho chip CNN và điều khiển hoạt động của chip này. Ngoài ra DSP đóng vai trò quan trọng khi nhiệm vụ xử lý ảnh chứa một số toán hạng logic. Sau khi toàn bộ quá trình tính toán tiền xử lý phức tạp (bao gồm một số lượng lớn toán hạng xử lý ảnh) được thực hiện bởi chip CNN, DSP sẽ hoàn thành nốt nhiệm vụ còn lại. Tức là, chíp CNN sẽ lọc ra khoảng 1% ảnh cần quan tâm, và DSP sẽ chỉ làm việc trên phần dữ liệu được rút gọn đáng kể này. Hai bộ xử lý chất lượng cao được tích hợp để tạo ra một hệ thống thị giác cực mạnh tựa sinh học, có khả năng tính toán ảnh thời gian thực trong các ứng dụng có yêu cầu cao. Bi-I cũng có một bộ xử lý truyền thông hỗ trợ các giao diện khác nhau, trong đó, giao diện quan trọng nhất là Ethernet 100 Mbit. Chương trình chạy trên Bi-i được nạp qua Ethernet và máy tính chủ có thể đọc, ghi từ Bi-i qua Ethernet. Ứng dụng công nghệ mạng nơ ron tế bào trong xử lý ảnh tốc độ cao (tốc độ xử lý trên 10000 ảnh/giây) trong công nghiệp đã được nhiều nhóm nghiên cứu trên thế giới triển khai. Ở Việt Nam, lĩnh vực này còn mới mẻ và chưa được nghiên cứu nhiều.

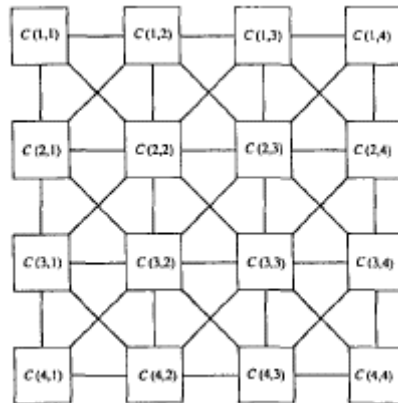
Báo cáo này nhằm giới thiệu cấu trúc, các tính chất cơ bản của mạng nơ ron tế bào, máy tính thị giác Bi-I, các kết quả nghiên cứu đã đạt được và xu thế phát triển của CNN trong giai đoạn tới. Báo cáo cũng điếm qua các khả năng ứng dụng của công nghệ CNN trong công nghiệp, trong các lĩnh vực y tế, an ninh và quốc phòng.

Mạng nơ ron tế bào CNN

Máy tính số đang tiến dần đến giới hạn vật lý về tốc độ và kích thước. Để vượt qua các trở ngại này một loại công nghệ tính toán mới dạng "mạng nơron" đã được đưa ra trên cơ sở chứa một vài cấu trúc của mạng nơron sinh học và được thực hiện trong các mạch điện tích hợp. Đặc điểm mấu chốt của mạng nơron tế bào là xử lý song song không đồng bộ, động học thời gian liên tục và ảnh hưởng toàn cục của các phần tử mạng.

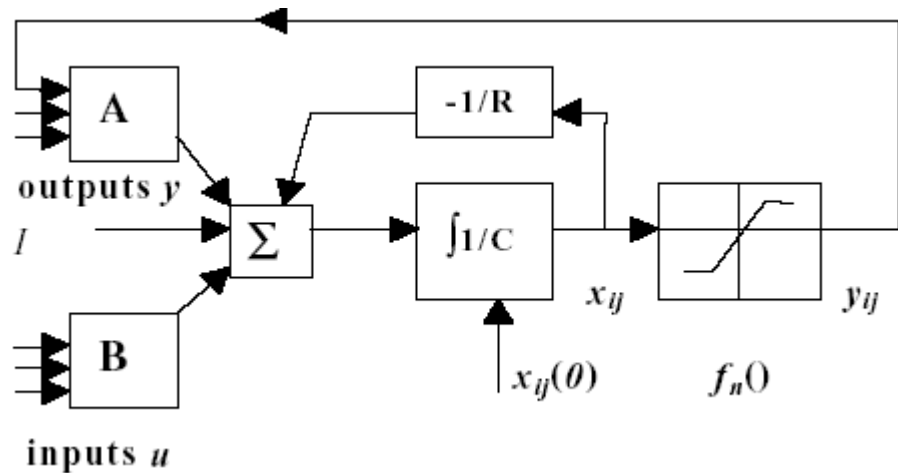
CNN được Leon O. Chua và L. Yang giới thiệu năm 1988 [1] [4]. Tư tưởng chung là sử dụng một mảng đơn giản các tế bào liên kết nhau cục bộ để xây dựng một hệ thống xử lý tín hiệu analog mạnh.

Khối mạch cơ bản của CNN được gọi là tế bào (tế bào). Nó chứa các phần tử mạch tuyến tính và phi tuyến bao gồm các tụ tuyến tính, các điện trở tuyến tính, các nguồn điều khiển tuyến tính và phi tuyến, và các nguồn độc lập. Mỗi một tế bào trong CNN chỉ nối tới các tế bào láng giềng của nó. Các tế bào liền kề có thể ảnh hưởng trực tiếp lẫn nhau. Các tế bào không có kết nối trực tiếp có thể tác động đến nhau bởi tác động lan truyền của hệ động lực liên tục của mạng CNN. Một ví dụ CNN 2 chiều được xem trong Hình 1.



Hình 1. Mạng CNN hai chiều

Về lý thuyết có thể định nghĩa một mạng CNN có nhiều chiều, nhưng ở đây chúng ta tập trung trong trường hợp mạng CNN hai chiều cho bài toán xử lý ảnh nhanh. Các kết quả có thể suy diễn dễ dàng trong trường hợp mạng lớn hơn 2 chiều. Hệ động lực của một tế bào của mạng CNN có thể mô tả trong Hình 2.

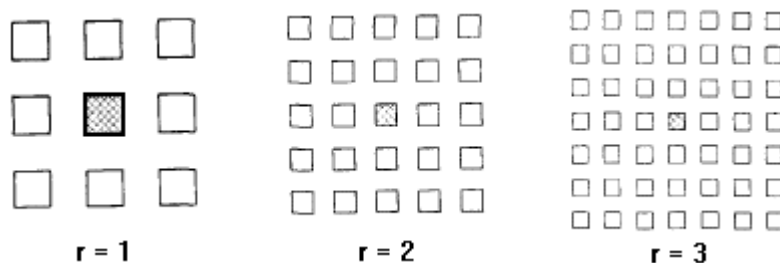


Hình 2. Sơ đồ khối của một tế bào CNN

Khảo sát một mạng CNN $M \times N$ có $M \cdot N$ tế bào sắp xếp trong M hàng và N cột. Chúng ta gọi tế bào trong hàng i và cột j là tế bào (i,j) và ký hiệu là $C(i,j)$. Láng giềng r của tế bào $C(i,j)$ trong một mạng CNN được định nghĩa bởi

$$N_r(i, j) = \{C(k, l) | \max\{|k - i|, |l - j|\} \leq r, 1 \leq k \leq M; 1 \leq l \leq N\} \quad (1)$$

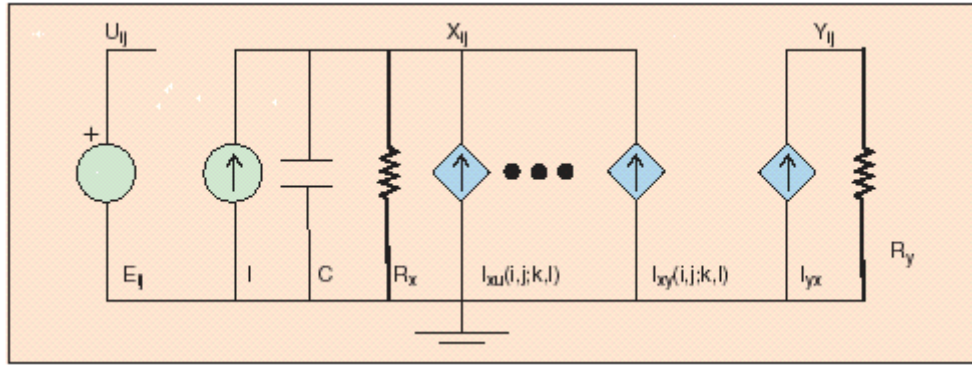
trong đó r là một số nguyên dương (Xem hình 3).



Hình 3. Láng giềng của tế bào $C(i,j)$ với $r=1$, $r=2$ và $r=3$

Thường thường chúng ta gọi láng giềng $r=1$ là "láng giềng 3×3 ", láng giềng $r=2$ là "láng giềng 5×5 ", láng giềng $r=3$ là "láng giềng 7×7 ". Dễ dàng nhận thấy rằng hệ thống láng giềng được định nghĩa như trên đưa ra là thuộc tính đối xứng trong ý nghĩa là nếu $C(i,j) \in N_r(k,l)$ thì $C(k,l) \in N_r(i,j)$ cho tất cả $C(i,j)$ và $C(k,l)$ trong một mạng CNN.

Một ví dụ tiêu biểu về mạch điện của một tế bào của mạng CNN $C(i,j)$ được mô tả trong Hình 4.



Hình 4. Sơ đồ mạch điện của một tế bào

trong đó u : là tín hiệu đầu vào

x : là tín hiệu trạng thái

y : là tín hiệu đầu ra

Nút điện áp v_{xij} của tế bào $C(i,j)$ là *trạng thái* của tế bào và điều kiện ban đầu của chúng được giả định là có độ lớn nhỏ hơn hoặc bằng 1.

Nút điện áp v_{uij} được gọi là *đầu vào* của tế bào và điều kiện ban đầu của chúng cũng được giả định là có độ lớn nhỏ hơn hoặc bằng 1.

Nút điện áp u_{yij} là *đầu ra*. Mạch điện cơ bản của một tế bào CNN gồm có:

- + Một nguồn dòng một chiều
- + Một tụ tuyến tính C
- + Hai điện trở tuyến tính R_x và R_y
- + Một cặp nguồn dòng được điều khiển qua 2 x m điện áp tuyến tính đến từ các tế bào láng giềng điện áp đầu vào v_{ukl} và hồi tiếp từ điện áp đầu ra v_{ykl} của mỗi một tế bào láng giềng $C(k,l)$; m bằng số tế bào láng giềng.

Đối với $C(k,l) \in N_r(i,j)$, $I_{xy}(i,j;k,l)$ và $I_{xu}(i,j;k,l)$ là nguồn dòng tuyến tính được điều khiển bằng điện áp với các đặc điểm:

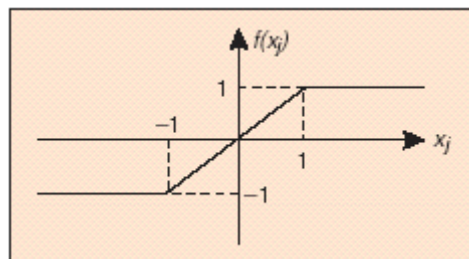
$$I_{xy}(i,j;k,l) = A(i,j;k,l)v_{ykl}$$

$$I_{xu}(i,j;k,l) = B(i,j;k,l)v_{ukl}$$

Chỉ có một phần tử phi tuyến trong mỗi một tế bào là phần dòng được điều khiển bằng nguồn áp:

$$I_{yx} = (1/R_y)f(v_{xyj})$$

Trong đó hàm $f(\cdot)$ là hàm bão hòa có đặc tính được mô tả trong hình 5.



Hình 5. Đặc tính đầu ra phi tuyến của tế bào

Động lực học của một tế bào CNN được mô tả bằng một hệ phương trình đồng nhất, bao gồm phương trình trạng thái, phương trình đầu vào, một số điều kiện ràng buộc, và một số thông số giả định.

Phương trình trạng thái:

$$C \frac{dv_{xij}(t)}{dt} = -\frac{1}{R_x} v_{xij}(t) + \sum_{C(k,l) \in N_r(i,j)} A(i,j;k,l) \cdot v_{ukl} + \sum_{C(k,l) \in N_r(i,j)} B(i,j;k,l) \cdot v_{ukl}(t) + I_{bias}$$

$$1 \leq i \leq M; 1 \leq j \leq N \quad (2a)$$

Phương trình đầu ra:

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

$$1 \leq i \leq M; 1 \leq j \leq N \quad (2b)$$

Phương trình đầu vào:

$$v_{uij}(t) = E_{ij} \quad 1 \leq i \leq M; 1 \leq j \leq N \quad (2c)$$

Các điều kiện ràng buộc (Constraint):

$$|v_{xij}(0)| \leq 1 \quad 1 \leq i \leq M; 1 \leq j \leq N \quad (2d)$$

$$|v_{uij}| \leq 1 \quad 1 \leq i \leq M; 1 \leq j \leq N \quad (2e)$$

Các thông số giả định (Assumption):

$$A(i,j;k,l) = A(k,l;i,j) \quad 1 \leq i,k \leq M \quad 1 \leq j,l \leq N \quad (2f)$$

$$C > 0, R_x > 0 \quad (2g)$$

Động lực học của các tế bào CNN có cả cơ chế liên hệ ngược (feedback) từ đầu ra và liên hệ thuận (feedforward) qua đầu vào điều khiển. Ảnh hưởng của đầu ra liên hệ ngược hồi tiếp phụ thuộc vào trọng số tương tác $A(i,j;k,l)$ và được coi là toán tử liên hệ ngược. Tác động của đầu vào phụ thuộc vào trọng số $B(i,j;k,l)$ được gọi là toán tử đầu vào.

Một số nhận xét chung:

a) Tất cả các tế bào(cell) bên trong của mạng nơ ron tế bào có cùng cấu trúc mạch và giá trị các phần tử trong mạch. Những tế bào bên trong này có $(2r+1)^2$ tế bào lân cận, trong đó r là vùng lân cận, đã định nghĩa ở trên. Các tế bào khác được gọi là tế bào bao quanh. Mạng nơ ron tế bào là tập hợp của các phương trình vi phân phi tuyến của các tế bào trong mạng.

b) Đặc tính động của mạng nơ ron tế bào bao gồm cả hai phần điều khiển đầu vào và phản hồi đầu ra. Kết quả phản hồi đầu ra phụ thuộc vào trọng số liên kết $A(i,j,k,l)$, kết quả của điều khiển đầu vào phụ thuộc vào trọng số $B(i,j,k,l)$. Do đó $A(i,j,k,l)$ được coi như toán tử phản hồi, $B(i,j,k,l)$ là toán tử điều khiển. Ta còn gọi A và B là các mẫu (template) của mạng. Ngoài ra ngưỡng I_{bias} còn được ký hiệu là hiệu dịch z_i cũng đóng vai trò quan trọng trong đặc tính động của mạng CNN.

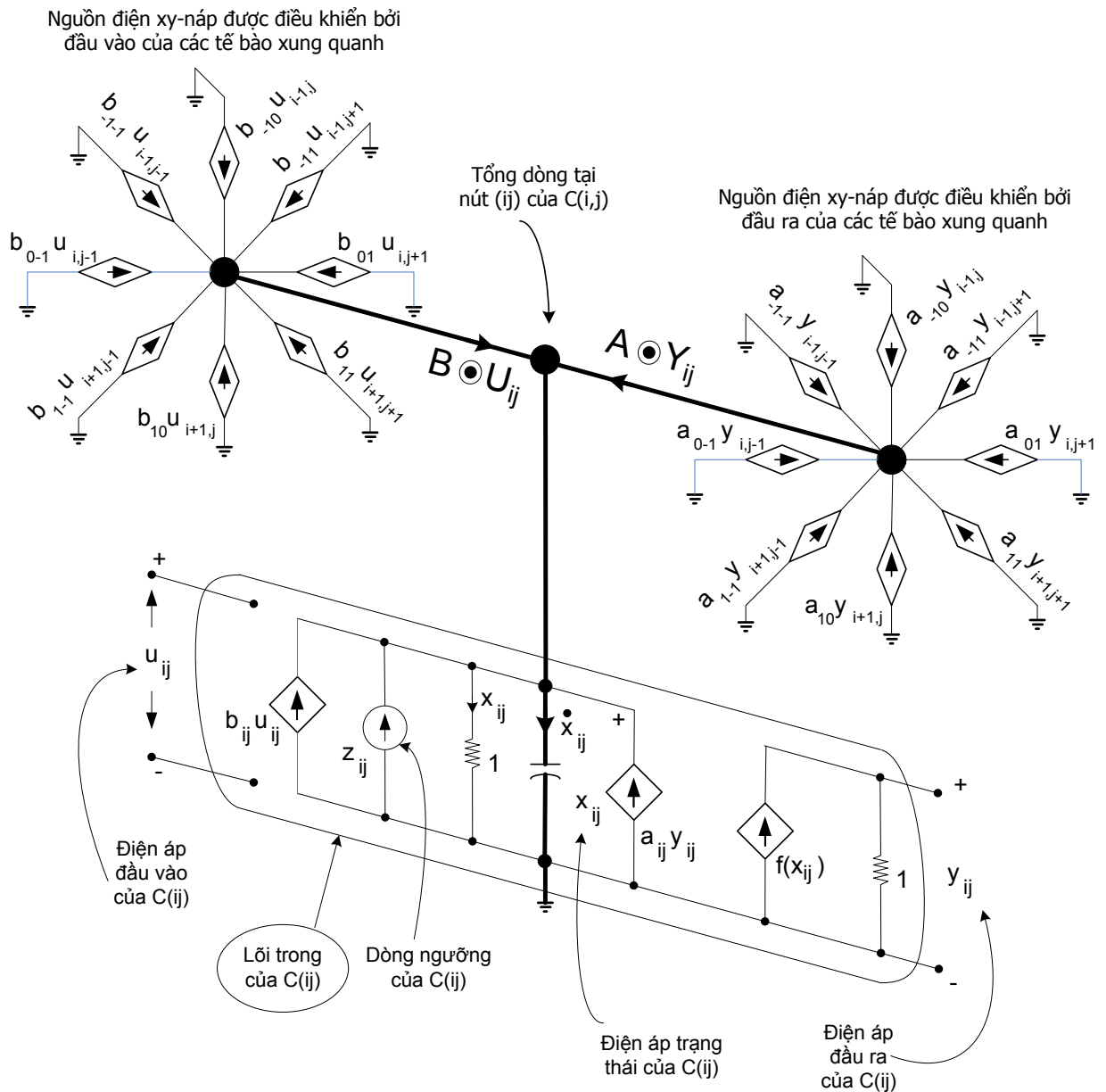
c) Giá trị của các phần tử trong mạch có thể được chọn tùy ý. Trong thực tế, R_x, R_y xác định năng lượng tiêu tán trong mạch và thường được chọn trong khoảng $1 \text{ k}\Omega$ tới $1 \text{ M}\Omega$. CR_x chính là hằng số thời gian của đặc tính động của mạch, và nó thường được chọn trong khoảng 10^{-8} tới 10^{-5} s.

Chức năng của một CNN sẽ được xác định hoàn toàn khi biết các mẫu phản hồi A , mẫu điều khiển B và hiệu dịch z . Với CNN tuyến tính bất biến không gian 3×3 đó là bộ 19 số thực sau

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, \quad B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, \quad z$$

- Đối với ứng dụng của CNN trong xử lý ảnh, đầu vào u_{kl} thường là cường độ điểm ảnh của ảnh ảnh xám kích thước $M \times N$ và ta quy định $-1 \leq u_{kl} \leq +1$ với màu trắng ký hiệu cho -1 và màu đen ký hiệu cho $+1$. Các biến khác cũng có thể xem như là các ảnh.
- Trong trường hợp tổng quát, A , B và z có thể thay đổi theo ij và thời gian t . Tuy nhiên trong nhiều bài toán, người ta coi chúng bất biến theo thời gian và không gian.

Ta có thể trình bày một tế bào CNN với đầy đủ tác động phản hồi và điều khiển như Hình 6 sau [3]:



Hình 6 – Sơ đồ tín hiệu một tế bào CNN

Dải đặc tính động của mạng nơ ron tế bào

Trước khi thiết kế mạng nơ ron tế bào, ta cần xác định dải đặc tính động (dynamics range) để xem nó có thỏa mãn giả thiết rằng buộc của mạng hay không. Vấn đề này đã được chứng minh trong [1] như sau:

Tất cả các trạng thái v_{xyj} của mỗi tế bào trong mạng nơ ron tế bào được giới hạn tại mọi thời điểm $t > 0$ và với bất kỳ mạng nơ ron tế bào nào, v_{max} có thể được tính theo công thức sau:

$$v_{max} = 1 + R_x |I| + R_x \max_{1 \leq i \leq M, 1 \leq j \leq N} \left[\sum_{C(k,l) \in N_i(i,j)} (|A(i,j;k,l)| + |B(i,j;k,l)|) \right]. \quad (3)$$

Các thông số R_x , C , I , $A(i,j,k,l)$ và $B(i,j,k,l)$ là những hằng số có giới hạn, vì vậy trạng thái của các tế bào có giới hạn trên là v_{max} và có thể tính theo công thức (3)

Lưu ý:

Trong thiết kế mạch thực tế, ta dễ dàng chọn dải cho các thông số trong mạch để

$$R_x |I| \approx 1; R_x |A(i,j,k,l)| \approx 1; R_x |B(i,j,k,l)| \approx 1 \text{ với mọi } i, j, k, l$$

Ở đây, chúng ta có thể dễ dàng ước lượng giới hạn trên của dải đặc tính động của mạng nơ ron tế bào. Ví dụ nếu vùng lân cận của mạng nơ ron tế bào là 3×3 thì chúng ta có thể có $v_{max} \approx 20V$, là giá trị trong phạm vi dải nguồn cung cấp thông thường cho các mạch IC CMOS.

Trạng thái ổn định của mạng nơ ron tế bào.

Một ứng dụng quan trọng của mạng nơ ron tế bào là xử lý ảnh. Chức năng cơ bản của mạng nơ ron tế bào trong quá trình xử lý ảnh là ánh xạ hoặc chuyển hình ảnh đầu vào tương ứng với hình ảnh đầu ra. Ở đây, chúng ta hạn chế hình ảnh đầu ra là ảnh nhị phân với giá trị điểm ảnh là -1 và 1 . Tuy nhiên hình ảnh đầu vào có thể có nhiều mức xám, cung cấp điện áp tương ứng thỏa mãn (2e). Điều này cho thấy mạng nơ ron tế bào xử lý ảnh luôn luôn phải hội tụ về trạng thái ổn định là hằng số sau một thời gian quá độ ngắn khi có ảnh đầu vào.

Để đảm bảo sự hội tụ của mạng nơ ron tế bào ta cần xác định các điều kiện hoặc giới hạn cần thiết cho sự hội tụ. Ta sẽ khảo sát đặc tính hội tụ và các vấn đề liên quan đến tính hội tụ của mạng nơ ron tế bào.

Một trong những kỹ thuật hiệu quả cho phân tích đặc tính hội tụ của hệ động lực phi tuyến là phương pháp ổn định Lyapunov. Chúng ta sẽ định nghĩa hàm Lyapunov cho mạng nơ ron tế bào như sau:

$$E(t) = -\frac{1}{2} \sum_{(i,j)} \sum_{(k,l)} A(i,j;k,l) v_{yij}(t) v_{ykl}(t) + \frac{1}{2R_x} \sum_{(i,j)} v_{yij}(t)^2 - \sum_{(i,j)} \sum_{(k,l)} B(i,j;k,l) v_{yij}(t) v_{ukl} - \sum_{(i,j)} I v_{yij}(t). \quad (4)$$

Nhận xét:

a) Quan sát thấy rằng, hàm Lyapunov $E(t)$ ở trên là hàm phụ thuộc vào đầu vào v_u , và đầu ra v_y , là các giá trị điện áp của mạch điện. Mặc dù $E(t)$ không có toàn bộ thông tin về biến trạng thái v_{xij} , nhưng chúng ta cũng có thể nhận được trạng thái ổn định của các biến trạng thái từ đặc tính $E(t)$.

b) Hàm Lyapunov $E(t)$ định nghĩa ở trên có thể hiểu như một hàm “năng lượng” của mạng nơ ron tế bào, mặc dù ý nghĩa vật lý chính xác của nó không được rõ ràng cho lắm. Như ở định lý 2 sau đây sẽ cho thấy $E(t)$ luôn luôn hội tụ tới điểm cực tiểu, mà mạng nơ ron tế bào tạo ra đầu ra theo yêu cầu.

Hàm $E(t)$ được định nghĩa ở (4) được giới hạn [1] bởi

$$\max_t |E(t)| \leq E_{\max} \quad (5a)$$

Trong đó:

$$E_{\max} = \frac{1}{2} \sum_{(i,j)} \sum_{(k,l)} |A(i,j;k,l)| + \sum_{(i,j)} \sum_{(k,l)} |B(i,j;k,l)| + MN \left(\frac{1}{2R_x} + |I| \right) \quad (5b)$$

Bằng việc lấy đạo hàm phương trình (4) với biến thời gian t , và sau khi biến đổi ta có

$$\frac{dE(t)}{dt} = - \sum_{|v_{xij}| < 1} C \left[\frac{dv_{xij}(t)}{dt} \right]^2 \leq 0. \quad \square \quad (6)$$

Hoặc nói cách khác ta có hàm $E(t)$ được là một hàm giảm đều.

Như vậy với bất kỳ đầu vào v_u và trạng thái ban đầu v_x của mạng nơ ron tế bào, chúng ta sẽ có:

$$\lim_{t \rightarrow \infty} E(t) = \text{constant} \quad (7a)$$

$$\lim_{t \rightarrow \infty} \frac{dE(t)}{dt} = 0. \quad (7b)$$

Sau khi mạng nơ ron tế bào ổn định, chúng luôn đạt được đầu ra dc là hằng số. Nói một cách khác ta có:

$$\lim_{t \rightarrow \infty} v_{yij}(t) = \text{constant}, \quad 1 \leq i \leq M; 1 \leq j \leq N \quad (8a)$$

hoặc

$$\lim_{t \rightarrow \infty} \frac{dv_{yij}(t)}{dt} = 0, \quad 1 \leq i \leq M; 1 \leq j \leq N. \quad (8b)$$

Theo [1] nếu thông số của mạch thỏa mãn điều kiện:

$$A(i,j;i,j) > \frac{1}{R_x}, \quad (9)$$

thì mỗi tế bào của mạng nơ ron tế bào sẽ đi đến điểm cân bằng bền sau một thời gian quá độ. Tuy nhiên, biên độ của tất cả các điểm cân bằng bền đều lớn hơn 1, nói cách khác ta có:

$$\lim_{t \rightarrow \infty} |v_{xij}(t)| > 1, \quad 1 \leq i \leq M; 1 \leq j \leq N \quad (10a)$$

$$\lim_{t \rightarrow \infty} v_{yij}(t) = \pm 1, \quad 1 \leq i \leq M; 1 \leq j \leq N. \quad (10b)$$

Nhận xét:

- a) Vấn đề này rất có ý nghĩa cho mạng nơ ron tế bào bởi vì nó bảo đảm mạch điện sẽ không dao động hoặc rơi vào trạng thái hỗn loạn.
- b) Phương trình (10) bảo đảm mạng nơ ron tế bào có đầu ra là giá trị nhị phân. Đặc tính này quyết định cho việc giải quyết bài toán phân loại trong ứng dụng xử lý ảnh.
- c) Từ $A(i, j, k, l)$ tương ứng với phản hồi từ đầu ra của tế bào $C(i, j)$ tới đầu vào của nó, điều kiện (9) quy định số nhỏ nhất các tín hiệu phản hồi dương để chắc chắn rằng trạng thái đầu ra ổn định của mỗi tế bào là +1 hoặc -1. Chú ý rằng, điều kiện này luôn luôn bị xâm phạm ở mạng nơ ron Hopfield ngay cả khi tất cả các hệ số đường chéo là 0. Để bảo đảm đầu ra nhị phân đồng dạng ± 1 thì cần phải chọn độ dốc lớn lý tưởng (vô tận) trong vùng tuyến tính của hàm phi tuyến $f(.)$ như trong hình 4. Slope tương ứng trong mạng nơ ron tế bào luôn luôn được chọn bằng 1.

Ví dụ mô phỏng mạng nơ ron tế bào đơn giản.

Ta khảo sát một ví dụ đơn giản để minh họa mạng nơ ron tế bào hoạt động thế nào.

Mạng nơ ron tế bào lấy ví dụ trong mục này được cho trong hình 1. Kích thước mạng là 4×4 , Các phần tử trong mạch của tế bào $C(i, j)$ được chọn như sau:

Với bất kỳ $C(k, l) \in N_R(i, j)$ và $r=1$ cho các trọng liên kết 3×3 (xem hình 2) có số liệu như sau:

$$\begin{aligned} A(i, j; i-1, j-1) &= 0; \\ A(i, j; i-1, j) &= 10^{-3} \Omega^{-1}; \\ A(i, j; i-1, j+1) &= 0 \\ A(i, j; i, j-1) &= 10^{-3} \Omega^{-1}; \\ A(i, j; i, j) &= 2.0 \times 10^{-3} \Omega^{-1}; \\ A(i, j; i, j+1) &= 10^{-3} \Omega^{-1} \\ A(i, j; i+1, j-1) &= 0; \\ A(i, j; i+1, j) &= 10^{-3} \Omega^{-1}; \\ A(i, j; i+1, j+1) &= 0 \\ C &= 10^{-9} \text{ F}; R_x = 10^3 \Omega; I = 0; \\ B(i, j; k, l) &= 0, \quad \text{for } C(k, l) \in N_1(i, j). \end{aligned}$$

Với $B(i, j, k, l) = 0$, những hệ số 3×3 $A(i, j, k, l)$ xác định quỹ đạo của mạng nơ ron tế bào trong thời gian. Chúng ta có thể sắp xếp các hệ số theo dạng ma trận và nó được gọi là mẫu (cloning template) thể hiện đặc tính động của mạng nơ ron tế bào như sau:

$$\begin{array}{cc} \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ 1.0 & 2.0 & 1.0 \\ 0.0 & 1.0 & 0.0 \end{bmatrix} & \begin{bmatrix} 0.0 & -0.25 & 0.0 \\ -0.25 & 2.0 & -0.25 \\ 0.0 & -0.25 & 0.0 \end{bmatrix} \\ \text{(a)} & \text{(b)} \end{array}$$

Đơn vị của những mẫu liên kết ở đây là $10^{-3} \Omega^{-1}$

Phương trình động lực của mạng nơ ron tế bào tương ứng với các thông số ở trên có dạng như sau:

$$\begin{aligned} \frac{dv_{xij}(t)}{dt} = & 10^6 \left[-v_{xij}(t) + 2v_{yij}(t) \right. \\ & \left. + v_{yij-1}(t) + v_{yij+1}(t) + v_{yi+1j}(t) + v_{yij+1}(t) \right] \end{aligned} \quad (11a)$$

và

$$v_{yij}(t) = 0.5 \left(|v_{xij}(t) + 1| - |v_{xij}(t) - 1| \right), \quad \text{for } 1 \leq i \leq 4; 1 \leq j \leq 4. \quad (11b)$$

Để thuận tiện phương trình (11a) có thể viết như sau:

$$\frac{dv_{xij}(t)}{dt} = -10^6 v_{xij}(t) + 10^6 \underbrace{\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}}_T \otimes v_{yij}(t) \quad (12)$$

Ở đây toán tử tích chập 2-D (two-dimensional convolution operator) \otimes được định nghĩa như sau:

$$T \otimes v_{yij} = \sum_{C(k,l) \in N(i,j)} T(k-i, l-j) v_{kl} \quad (13)$$

Trong đó $T(m, n)$ chỉ rõ hàng thứ m và cột thứ n của ma trận T , $m = -1, 0, 1$ và $n = -1, 0, 1$.

Ở định nghĩa trên $A(i, j, k, l)$ chỉ phụ thuộc vào i và j của mạng nơ ron tế bào. Đặc tính này được gọi là hệ bất biến theo không gian (space invariant), và nó gợi ý rằng $A(i, j, k, l)$ có thể biểu đạt như $A(k-i, l-j)$. Điều này cho phép chúng ta chỉ rõ đặc tính động (dynamic rules) của mạng nơ ron tế bào bằng việc sử dụng ma trận liên kết.

Để xác định trạng thái trong thời gian quá độ của (11a), chúng ta cho điện áp khởi đầu $v_{xij}(0)$ qua tụ điện của mỗi tế bào $C(i, j)$. Mỗi giá trị điện áp khởi đầu có thể được chỉ định điện áp bất kỳ nào giữa -1 và 1 như quy định trong (2d).

Mô phỏng mạch điện này chúng ta có thể nhận được đáp ứng trong thời gian quá độ ngắn của mạng CNN.

Ta mô phỏng tính xử lý nhanh của mạng nơ ron tế bào ở trên với điều kiện ban đầu như trong các mảng ở Hình 7a [1]. Biến trạng thái của mạch v_x tại thời điểm $t = 5 \mu\text{s}$ nhận được như trong hình 7(b). Giá trị tuyệt đối lớn nhất của biến trạng thái tại $t = 5 \mu\text{s}$ bằng 6. Giới hạn cao hơn v_{max} của v_x được tính toán từ phương trình (3) của định lý 1 bằng 7.

-0.8	1.0	-1.0	-0.6
1.0	1.0	1.0	-1.0
-1.0	0.9	-1.0	-0.8
-0.9	-1.0	-0.7	-0.8

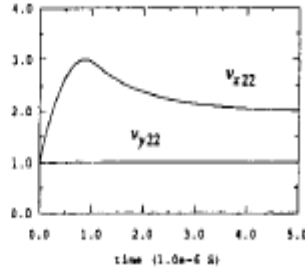
(a)

3.95	2.98	-2.97	-3.98
2.98	2.02	-3.94	-4.96
-2.98	-3.94	-5.94	-4.97
-3.98	-4.96	-4.97	-3.98

(b)

1.0	1.0	-1.0	-1.0
1.0	1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0

(c)



(d)

Hình 7. Mô phỏng với mạng nơ ron tế bào 4x4.

- Giá trị ban đầu của các biến trạng thái.
- Giá trị ổn định của các biến trạng thái
- Giá trị ổn định của các đầu ra
- Quỹ đạo của mạch tế bào C(2,2)

Đầu ra tương ứng v_y tại thời điểm $t = 5 \mu\text{s}$ như trong Hình 7(c). Quan sát thấy rằng tất cả các biến đầu ra là biến nhị phân, là 1 hoặc -1, như dự đoán (điều kiện $A(i, j, k, l) > 1/R_x$ là thỏa mãn).

Quá trình quá độ của trạng thái của một tế bào C(2,2) được mô tả trong hình 7(d). Giá trị khởi đầu của biến trạng thái bằng 1.0 và giá trị tại thời điểm $t = 5 \mu\text{s}$ bằng 2.02. Giá trị lớn nhất của $v_{x22}(t)$ bằng 3 và xảy ra xấp xỉ tại $t = 0.8 \mu\text{s}$. Khi biến trạng thái giữ trên 1.0 thì đầu ra tương ứng luôn là hằng số và bằng 1 như hình 4.

Trạng thái cân bằng ổn định tế bào (stable cell equilibrium state) v_{xij}^* của một tế bào chuẩn trong mạng nơ ron tế bào được định nghĩa như biến trạng thái v_{xij} của tế bào C(i, j), và nó thỏa mãn:

$$\left. \frac{dv_{xij}(t)}{dt} \right|_{v_{xij} = v_{xij}^*} = 0 \text{ and } |v_{xij}^*| > 1 \quad (14)$$

Trạng thái cân bằng ổn định tế bào thực sự đạt được bởi mỗi tế bào, phụ thuộc vào trạng thái ban đầu của nó và của những tế bào lân cận nó.

Sau đây chúng ta sẽ tập trung vào đặc tính động chung của mạng CNN chuẩn.

“Điểm cân bằng ổn định hệ thống của mạng nơ ron tế bào được định nghĩa là 1 véc tơ trạng thái với tất cả những thành phần của nó bao gồm những trạng thái cân bằng ổn định của mỗi tế bào”

Với định nghĩa trên cho thấy rằng mạng nơ ron tế bào chuẩn luôn luôn ở tại một điểm cân bằng *ổn định hệ thống* sau thời gian ngắn rời khỏi trạng thái 0. Đáp ứng của mạng nơ ron tế bào đơn giản là quỹ đạo đi từ một vài điểm khởi đầu và kết thúc tại điểm cân bằng của hệ thống. Như vậy bất kỳ điểm cân bằng *ổn định hệ thống* của mạng nơ ron tế bào là điểm tới hạn của tập các quỹ đạo của đáp ứng phương trình vi phân (2), giống như điểm tới hạn hấp dẫn có vùng hấp dẫn, hay tập hợp các quỹ đạo hội tụ về điểm này. Vì thế, không gian trạng thái của mạng nơ ron tế bào có thể được phân cắt bằng tập trong vùng lòng chảo tập trung quanh điểm cân bằng *ổn định hệ thống*.

Sự ổn định của mạng nơ ron tế bào chuẩn được chứng minh bằng toán học chặt chẽ trong các tài liệu [2], [3].

Tóm lại, mạng nơ ron tế bào xử lý tín hiệu bằng việc ánh xạ chúng từ một không gian tín hiệu này sang một không gian khác. Mạng nơ ron tế bào có thể ánh xạ trạng thái khởi tạo ban đầu của hệ thống tới một trong nhiều điểm cân bằng *ổn định hệ thống* riêng biệt. Nếu trạng thái ban đầu là $[-1.0, 1.0]^{M \times N}$ và không gian đầu ra là $[-1, 1]^{M \times N}$, thì ánh xạ động F có thể được định nghĩa như sau:

$$F: [-1.0, 1.0]^{M \times N} \rightarrow \{-1, 1\}^{M \times N} \quad (15)$$

điều này có nghĩa là ánh xạ F có thể sử dụng như phân cắt không gian tín hiệu liên tục sang vùng hấp dẫn của những điểm cân bằng *ổn định hệ thống* qua xử lý động.

Đặc tính động của mạng nơ ron tế bào với toán tử điều khiển $B(i, j, k, l) = 0$ và toán tử phản hồi khác không $A(i, j, k, l) \neq 0$ là sự gợi lại về hoạt động của ô tô mát tế bào 2D. Cả 2 chúng đều có khả năng xử lý tín hiệu song song và dựa trên những luật động tương tác lẫn nhau giữa tế bào và những tế bào lân cận gần nhất.

Sự khác nhau căn bản giữa mạng nơ ron tế bào và ô tô mát tế bào chính là đặc tính động của nó. Mạng CNN là hệ động lực thời gian liên tục, còn ô tô mát tế bào là hệ động lực rời rạc theo thời gian. Cả 2 hệ thống đều có những điểm tương tự nhau, chúng ta có thể sử dụng lý thuyết ô tô mát tế bào để khảo sát về hoạt động trạng thái *ổn định* của mạng nơ ron tế bào. Một điểm khác biệt giữa chúng là trong khi mạng nơ ron tế bào chuẩn sẽ luôn luôn đạt tới điểm cân bằng bền, còn ô tô mát tế bào luôn luôn kèm với các quá trình động phong phú hơn như quá trình hỗn loạn, giao động tuần hoàn và những hiện tượng phức tạp khác.

Tuy nhiên chúng ta đã thuần hóa mạng nơ ron tế bào chuẩn bằng việc chọn hàm phi tuyến sigmoid. Nếu chúng ta chọn các đặc tính phi tuyến khác có thể có nhiều hiện tượng động học phức tạp sẽ xảy ra trong mạng nơ ron tế bào do tính chất phi tuyến của nó.

CNN bất biến không gian

Do các ứng dụng CNN chủ yếu chỉ sử dụng các CNN chuẩn bất biến không gian với các láng giềng 3×3 ($r = 1$) nên ta xem xét thêm vai trò của 2 toán tử A, B và ngưỡng z trong mạng CNN.

a. *Vai trò của toán tử phản hồi $A(i, j, k, l)$*

Nhờ tính bất biến không gian, ta có thể viết:

$$\sum_{C(k,l) \in S_r(i,j)} A(i, j, k, l) y_{ij} = \sum_{|k-i| \leq 1} \sum_{|l-j| \leq 1} A(k-i, l-j) y_{ij}$$

$$\begin{aligned}
&= a_{-1,-1}y_{i-1,j-1} + a_{-1,0}y_{i-1,j} + a_{-1,1}y_{i-1,j+1} \\
&+ a_{0,-1}y_{i,j-1} + a_{0,0}y_{i,j} + a_{0,1}y_{i,j+1} \\
&+ a_{1,-1}y_{i+1,j-1} + a_{1,0}y_{i+1,j} + a_{1,1}y_{i+1,j+1} \\
&= \sum_{k=-1}^1 \sum_{l=-1}^1 a_{k,l}y_{i+k,j+l}
\end{aligned} \tag{16}$$

Trong đó $a_{m,n}=A(m,n)$

$$\hat{=} \begin{array}{ccc} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{array} \otimes \begin{array}{ccc} y_{i-1,j-1} & y_{i-1,j} & y_{i-1,j+1} \\ y_{i,j-1} & y_{i,j} & y_{i,j+1} \\ y_{i+1,j-1} & y_{i+1,j} & y_{i+1,j+1} \end{array} = A \otimes Y_{ij}$$

Ở đây, ma trận A được gọi là mẫu phản hồi, và dấu \otimes là ký hiệu cho tổng các tích vô hướng. Trong toán rời rạc, phép toán này được gọi là “*spatial convolution*”. Ma trận 3x3 của Y_{ij} trong (16) có thể có được bằng cách di chuyển một mặt nạ mờ với một cửa sổ 3x3 vào vị trí (i,j) của ảnh đầu ra Y, nên nó được gọi là *ảnh đầu ra* ở vị trí C(i,j).

Một phần tử a_{kl} được gọi là trọng số trung tâm của A nếu và chỉ nếu (k,l)=(0,0). Ngược lại, nó được gọi là phần tử vòng ngoài.

Đôi khi, để tiện dụng, người ta viết lại A như sau:

$$A = A^0 + \bar{A}$$

$$A^0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_{00} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \bar{A} = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & 0 & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix} \tag{17}$$

Với A^0 và \bar{A} được gọi là các mẫu phần tử trung tâm và vòng ngoài.

b. *Vai trò của toán tử điều khiển B(i,j,k,l)*

Tương tự như trên, ta có thể viết:

$$\begin{aligned}
\sum_{C(k,l) \in S_r(i,j)} B(i,j,k,l)u_{kl} &= \sum_{|k-i| \leq 1} \sum_{|l-j| \leq 1} B(k-i,l-j)u_{kl} \\
&= \sum_{k=-1}^1 \sum_{l=-1}^1 b_{kl}u_{i+k,j+l}
\end{aligned} \tag{18}$$

$$\hat{=} \begin{array}{ccc} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{array} \otimes \begin{array}{ccc} u_{i-1,j-1} & u_{i-1,j} & u_{i-1,j+1} \\ u_{i,j-1} & u_{i,j} & u_{i,j+1} \\ u_{i+1,j-1} & u_{i+1,j} & u_{i+1,j+1} \end{array} = B \otimes U_{ij}$$

Trong đó, ma trận B được gọi là *feedforward* hoặc mẫu đầu vào. U_{ij} là ảnh đầu vào được đặt mặt nạ.

Ma trận B cũng có thể được biểu diễn thành:

$$B = B^0 + \bar{B} \quad (19)$$

$$B^0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & b_{00} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \bar{B} = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & 0 & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}$$

Với B^0 và \bar{B} được gọi là các mẫu *feedforward* trung tâm và vòng ngoài.

c. *Vai trò của ngưỡng z*

Theo phân tích ở trên, ta có thể viết lại phương trình trạng thái của tế bào CNN như sau

$$\dot{x}_{ij} = -x_{ij} + A \otimes \dot{Y}_{ij} + B \otimes U_{ij} + z \quad (20)$$

Hay đơn giản hoá thành dạng sau:

$$\dot{x}_{ij} = \underbrace{-x_{ij} + a_{00}f(x_{ij})}_{g(x_{ij})} + \underbrace{\bar{A} \otimes Y_{ij} + B \otimes U_{ij} + z}_{w_{ij}(t)} \quad (21)$$

Và hàm

$$h_{ij}(x_{ij}, w_{ij}) = g(x_{ij}) + w_{ij}(x_{ij}, t) \quad (22)$$

được gọi là hàm nhíp độ (rate function), $g(x_{ij})$ được gọi là thành phần điểm truyền động (DP – driving point) bởi vì nó liên quan chặt chẽ đến khái niệm của lý thuyết mạch phi tuyến, và

$w_{ij}(x_{ij}, t) = \bar{A} \otimes Y_{ij} + B \otimes U_{ij} + z$ được gọi là mức bù (*offset level*).

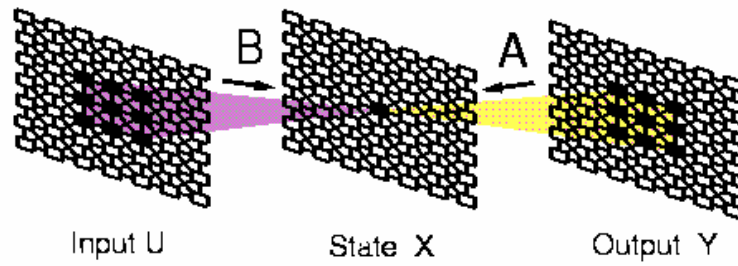
Ba lớp CNN đơn giản

Mỗi CNN được xác định duy nhất bởi ba mẫu $\{A, B, z\}$. Đây cũng chính là các trọng số liên kết của mạng nơ ron tế bào. Với CNN 3x3 ($r=1$) bộ ba này bao gồm 19 số thực. Do tập số thực là không đếm được nên có vô hạn mẫu CNN khác nhau, trong số chúng, 3 lớp CNN sau là đơn giản nhất và dễ tính toán nhất.

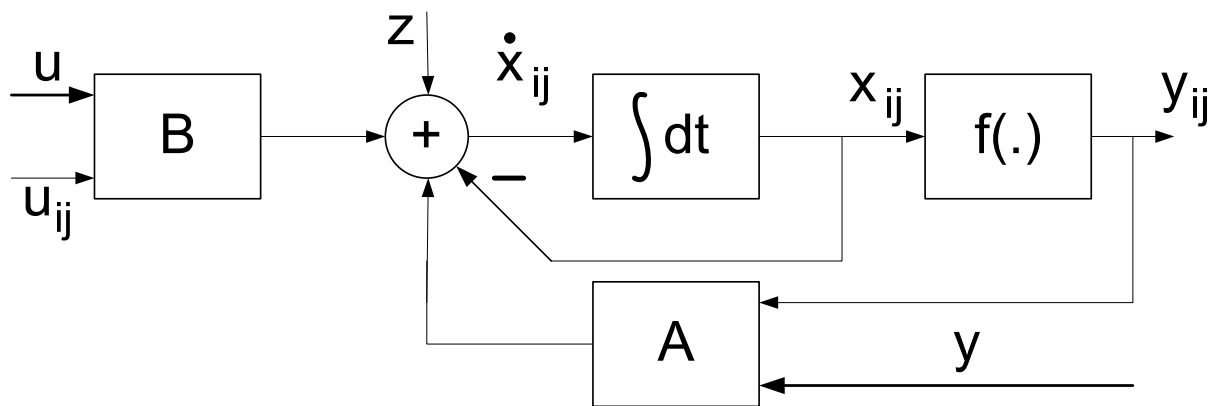
Trọng số liên kết kích thích và kiềm chế

Một trọng số liên kết phản hồi a_{kl} được gọi là kích thích (hoặc kiềm chế) nếu và chỉ nếu nó là dương (hoặc âm). Một trọng số liên kết là kích thích (hoặc kiềm chế) bởi vì nó làm cho hàm nhíp độ $h_{ij}(x_{ij}, w_{ij})$ ngày càng dương hơn (hoặc ít dương đi) đối với đầu vào dương, và từ đó tăng (hoặc giảm) giá trị \dot{x}_{ij} tức là tốc độ tăng của $x_{ij}(t)$. Một CNN bất biến không gian $\zeta(A, B, z)$ với láng giềng 3x3 có thể được biểu diễn theo sơ đồ sau

$$\zeta(A, B, z)$$



Dòng tín hiệu của CNN với láng giềng 3x3.



Sơ đồ cấu trúc của một tế bào $C(i,j)$

Hình 8 - CNN bất biến không gian $\zeta(A, B, z)$ với láng giềng 3x3

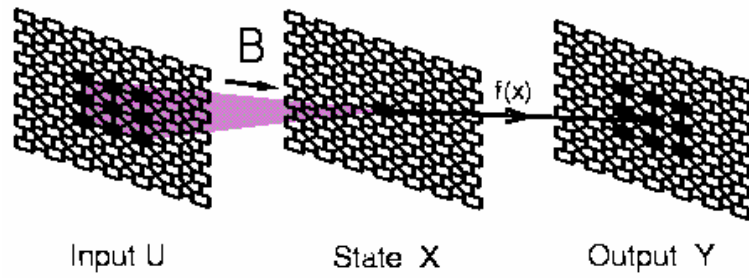
Lớp CNN phản hồi không (Zero-feedback, feedforward)

Một CNN thuộc lớp phản hồi không (Zero-feedback) $\zeta(0, B, z)$ nếu và chỉ nếu mọi phần tử của mẫu phản hồi đều là 0, tức là $A \equiv 0$.

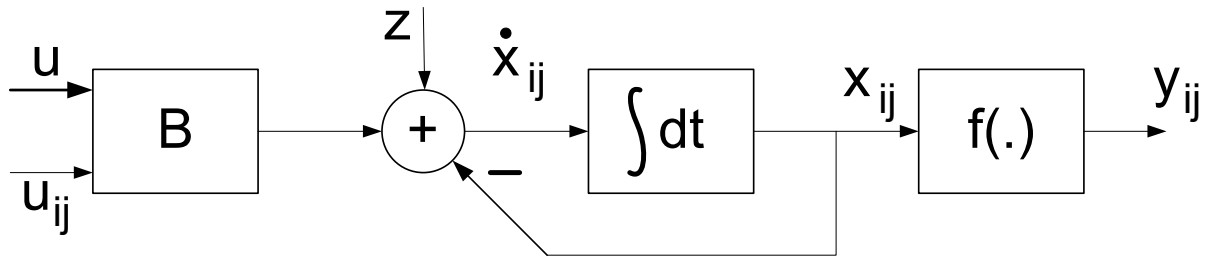
Mỗi tế bào của *CNN phản hồi không* được mô tả như sau:

$$\dot{x}_{ij} = -x_{ij} + B \otimes U_{ij} + z \quad (23)$$

Zero-Feedback CNN: $\xi(0, B, z)$



Cấu trúc dòng tín hiệu của CNN phản hồi không với láng giềng 3x3.



Hình 9 – Cấu trúc tế bào CNN phản hồi không (Zero-feedback)

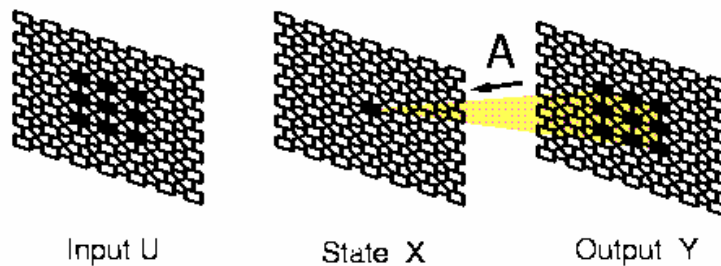
Lớp CNN đầu vào không (Zero-input, Autonomous) $\zeta(A, 0, z)$

Một CNN thuộc lớp *đầu vào không* (Zero-input) $\zeta(A, 0, z)$ nếu và chỉ nếu tất cả các thành phần của mẫu điều khiển đều là 0, tức $B \equiv 0$.

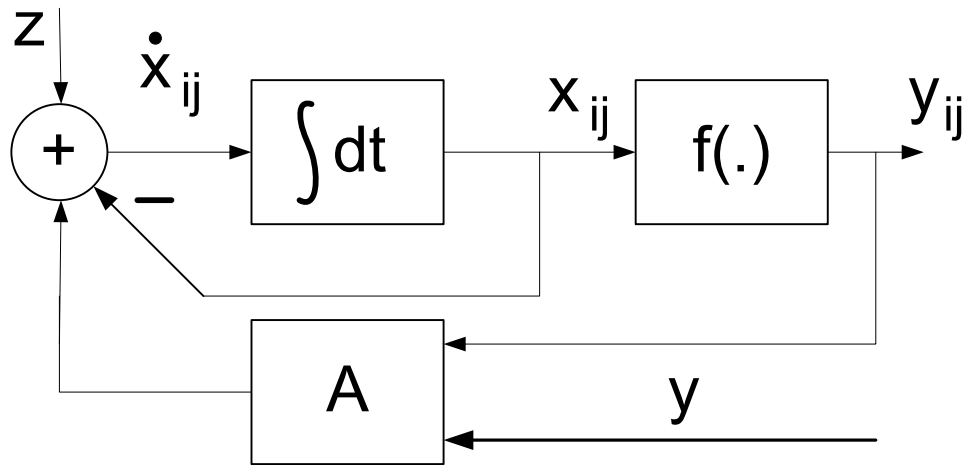
Mỗi tế bào của lớp này được mô tả như sau:

$$\dot{x}_{ij} = -x_{ij} + A \otimes Y_{ij} + z \quad (24)$$

Zero-Input CNN: $\xi(A, 0, z)$



Cấu trúc dòng mạng tín hiệu của một CNN đầu vào không với láng giềng 3x3



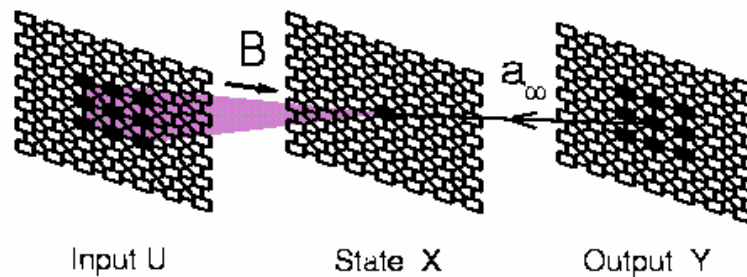
Hình 10 – Cấu trúc tế bào của CNN đầu vào không (Zero-input)

Lớp CNN không liên kết $\zeta(A^0, B, z)$

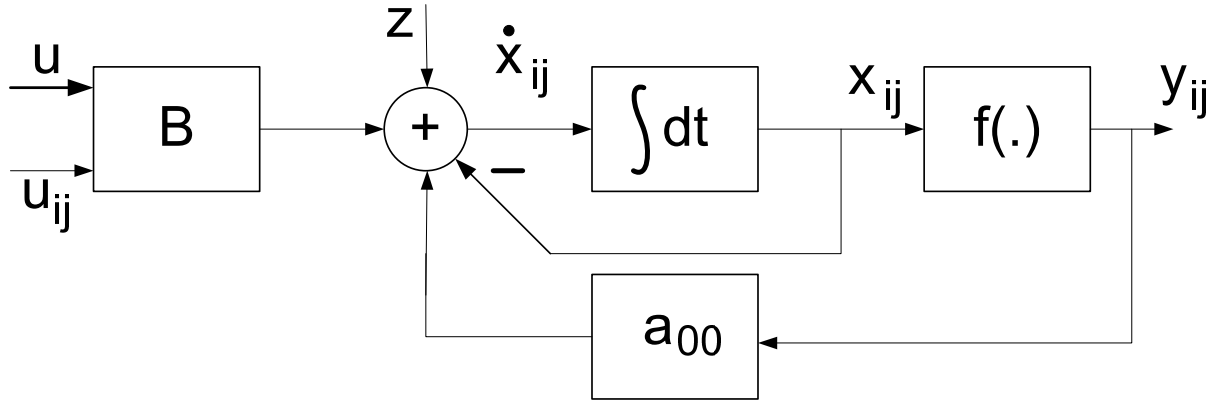
Một CNN thuộc về lớp không liên kết $\zeta(A^0, B, z)$ nếu và chỉ nếu $a_{ij} = 0$, $i \neq j$, tức là $\bar{A} \equiv 0$. Mỗi tế bào thuộc lớp CNN không liên kết được mô tả bởi một phương trình vi phân thường, phi tuyến, vô hướng mà không ghép với các láng giềng của nó:

$$\dot{x}_{ij} = -x_{ij} + a_{00} f(x_{ij}) + B \otimes U_{ij} + z \quad (25)$$

Uncoupled CNN: $\mathcal{C}(A^0, B, z)$



Cấu trúc dòng mảng dữ liệu của một CNN không liên kết với láng giềng 3x3.



Hình 11 - Cấu trúc tế bào của mạng CNN không liên kết

Động lực học của mạng nơ ron tế bào phi tuyến và có trễ

Trong một CNN tổng quát phi tuyến và có trễ, thay cho các nguồn dòng tuyến tính $I_{xy(i,j;k,l)}=A(i,j;k,l)V_{ykl}$ và $I_{xu(i,j;k,l)}=B(i,j;k,l)V_{ukl}$ trong tế bào C_{ij} là các nguồn dòng phi tuyến và trễ được định nghĩa như sau

$$\hat{A}_{ij;kl} (v_{ykl}, v_{yij}) + A_{ij;kl}^{\tau} v_{ykl} (t - \tau)$$

$$\hat{B}_{ij;kl} (v_{ukl}, v_{uij}) + B_{ij;kl}^{\tau} v_{ukl} (t - \tau)$$

Tức là thay vì có các mẫu A và B của các CNN chuẩn (trình bày trong phần 1) trước đây, chúng ta có các nguồn dòng phi tuyến và/ hoặc có trễ. Cấu trúc của tính chất phi tuyến trong các mẫu cũng rất quan trọng: đó là hàm của nhiều nhất là hai biến, cụ thể đó là điện áp đầu ra của tế bào C_{ij} và của tế bào lân cận. Ngoài ra điện áp đầu ra của tế bào có thể nhận khoản giá trị rộng hơn so với CNN chuẩn tức là $-K \leq V_{yij} \leq +K$ (thay vì nằm trong khoảng $-1, +1$).

Với các nguồn dòng phi tuyến, có trễ này, ta viết lại hệ phương trình (2) của CNN như sau:

- Phương trình trạng thái:

$$C \dot{v}_{xij} = -(1/R_x) v_{xij} (t) + I + \sum_{C_{kl} \in N_r(ij)} \hat{A}_{ij;kl} (v_{ykl} (t), v_{yij} (t))$$

$$+ \sum_{C_{kl} \in N_r(ij)} \hat{B}_{ij;kl} (v_{ukl} (t), v_{uij} (t)) \quad (26a)$$

$$+ \sum_{C_{kl} \in N_r(ij)} A_{ij;kl}^{\tau} v_{ykl} (t - \tau) + \sum_{C_{kl} \in N_r(ij)} B_{ij;kl}^{\tau} v_{ukl} (t - \tau)$$

Trong đó \hat{A} , \hat{B} và A^{τ} , B^{τ} tương ứng với các mẫu phi tuyến và trễ. $\hat{A}_{ij;k,l}$, $\hat{B}_{ij;k,l}$ là các hàm phi tuyến của nhiều nhất là hai biến. $A^{\tau}_{ij;k,l}$ và $B^{\tau}_{ij;k,l}$ là các số thực.

Ví dụ với $d_1=c_1(\exp(v_{ykl})-1)$, $d_2=c_2(v_{ykl}-V_{yij})$, các mẫu có thể được trình bày như sau:

$$\hat{A} = \begin{bmatrix} 0 & d_1 & 0 \\ d_1 & 2 & d_1 \\ 0 & d_1 & 0 \end{bmatrix} \quad \hat{B} = \begin{bmatrix} 0 & 0 & 0 \\ d_2 & 1 & d_2 \\ 0 & 0 & 0 \end{bmatrix}$$

- Phương trình đầu ra không nhớ:

$$v_{yij}(t) = f(v_{xij}(t)) \quad (26b)$$

Hàm $f(\cdot)$ là hàm tuyến tính từng đoạn tương tự như phương trình đầu ra của CNN chuẩn. Trong nhiều trường hợp người ta cho đầu ra có phương trình động lực riêng của mình:

$$\dot{v}_{yij} = -v_{yij} + f(v_{xij}(t)) \quad (26c)$$

Trong đó $f(x)$ có thể được tính như sau:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & 0 \leq x \leq 1 \\ 1, & x > 1 \end{cases}$$

- Phương trình đầu vào

$$v_{uij} = E_{ij} \quad (26d)$$

- Điều kiện

$$|v_{xij}(0)| \leq 1, \quad |v_{uij}| \leq 1 \quad (26e)$$

- Thông số

$$K \geq 0, C \geq 0, R_x > 0, \tau \geq 0 \quad (26f)$$

Hệ phương trình 26 là hệ phương trình của CNN phi tuyến, có trễ.

Chú ý: Trong trường hợp $\hat{A}_{ij,k,l}$, $\hat{B}_{ij,k,l}$ là các hàm tuyến tính (một biến) và $\tau=0$, hoặc

$$\begin{aligned} \hat{A}_{ij;kl} &= A_{ij;kl} v_{ykl} \\ \hat{B}_{ij;kl} &= B_{ij;kl} v_{ukl} \end{aligned} \quad (27)$$

và $A_{ij,k,l}^T = 0$ và $B_{ij,k,l}^T = 0$ ta nhận lại được hệ CNN chuẩn (tuyến tính bất biến không gian).

Giá trị giới hạn của CNN phi tuyến, có trễ

Với CNN được mô tả bởi các mẫu giới hạn (nhưng không giữ trễ) và đầu ra của tế bào được biểu diễn theo 26b (hàm tuyến tính từng đoạn không nhớ), tất cả các giá trị trạng thái V_{xij} bị giới hạn tại mọi thời điểm $t > 0$ và giá trị V_{max} có thể tính được theo công thức sau:

$$v_{max} = 1 + R_x |I| + R_x K \max \left[\sum_{C_{kl} \in N_r(ij)} \left(\max |\hat{A}_{ij;kl}| + \max |\hat{B}_{ij;kl}| \right) \right] \quad (28)$$

$$1 \leq i \leq M, 1 \leq j \leq N$$

Định đề này và một số ví dụ về khảo sát tính ổn định của hệ CNN phi tuyến có trễ được chứng minh trong [3].

Hỗn loạn trong mạng nơ ron tế bào [2]

Trong đa số các ứng dụng CNN, các mẫu phản hồi và điều khiển thường không phụ thuộc vào trạng thái ban đầu, đầu vào và điều kiện biên. Sau thời gian quá độ trạng thái và đầu ra hội tụ về một giá trị cân bằng dc nào đó. Những CNN như thế được coi như ổn định hoàn toàn.

Tuy nhiên, *không phải mọi* mẫu CNN đều dẫn đến sự ổn định. Trên thực tế, một vài mẫu CNN sẽ dẫn tới trạng thái *dao động theo chu kỳ*, một số khác thậm chí còn có hiện tượng chuyển tiếp không ngừng (không theo chu kỳ) gọi là hỗn độn (*chaos*).

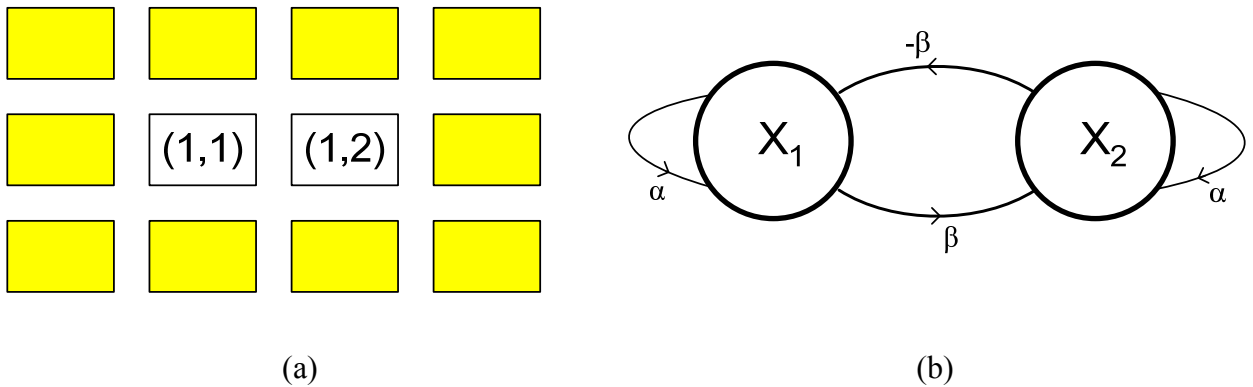
Các ứng dụng CNN hiện nay đa số đều cần đến giá trị đầu ra ổn định, tuy nhiên rất có thể sau này người ta sẽ khai thác những tiềm năng to lớn của các loại CNN hoạt động hỗn độn và dao động mà nay còn chưa được sử dụng. Ta hãy khảo sát tính dao động và hỗn độn của mạng CNN qua một số ví dụ sau:

Ví dụ về CNN 2 tế bào dao động

Xét một CNN 2 tế bào đặc trưng bởi điều kiện biên 0 với mẫu:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ \beta & \alpha & -\beta \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0 \quad (29)$$

CNN (M=1,N=2) này có trọng số liên kết phản hồi $a_{0,-1} = \beta$, $a_{0,0} = \alpha$, $a_{0,1} = -\beta$ và có thể biểu diễn bởi đồ thị luồng tín hiệu trong hình 12



Hình 12: a) 1x2 CNN cùng các tế bào biên với điều kiện biên 0,

$$y_{00} = y_{01} = y_{02} = y_{03} = y_{10} = y_{13} = y_{20} = y_{21} = y_{22} = y_{23} = 0.$$

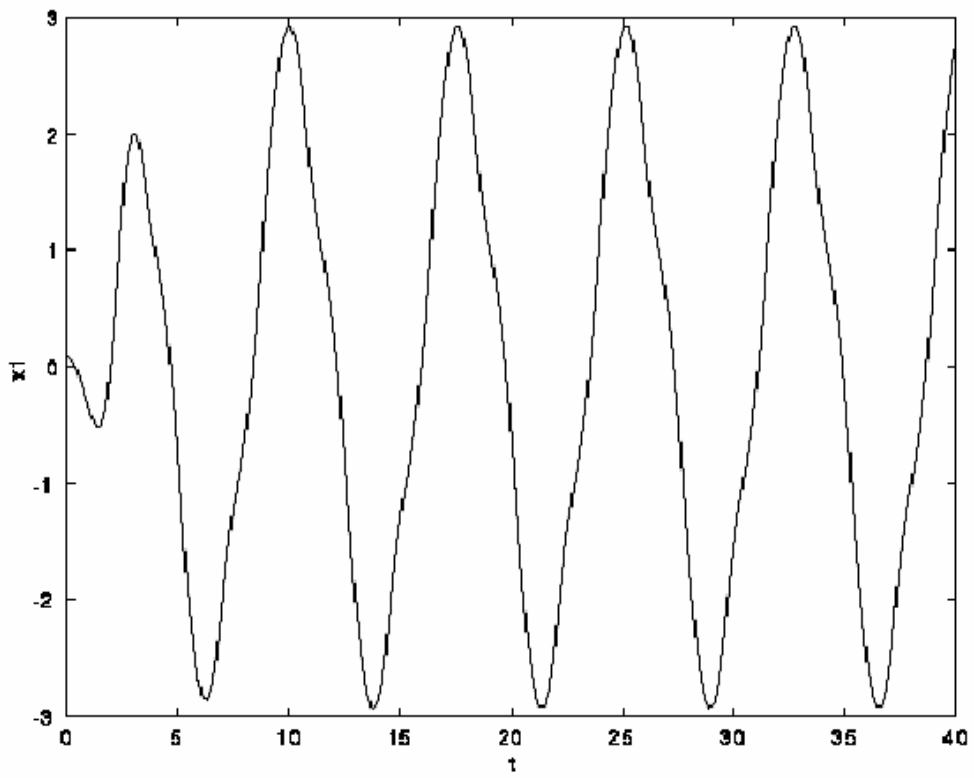
b) Đồ thị luồng dữ liệu tương ứng.

Phương trình trạng thái cho CNN này là:

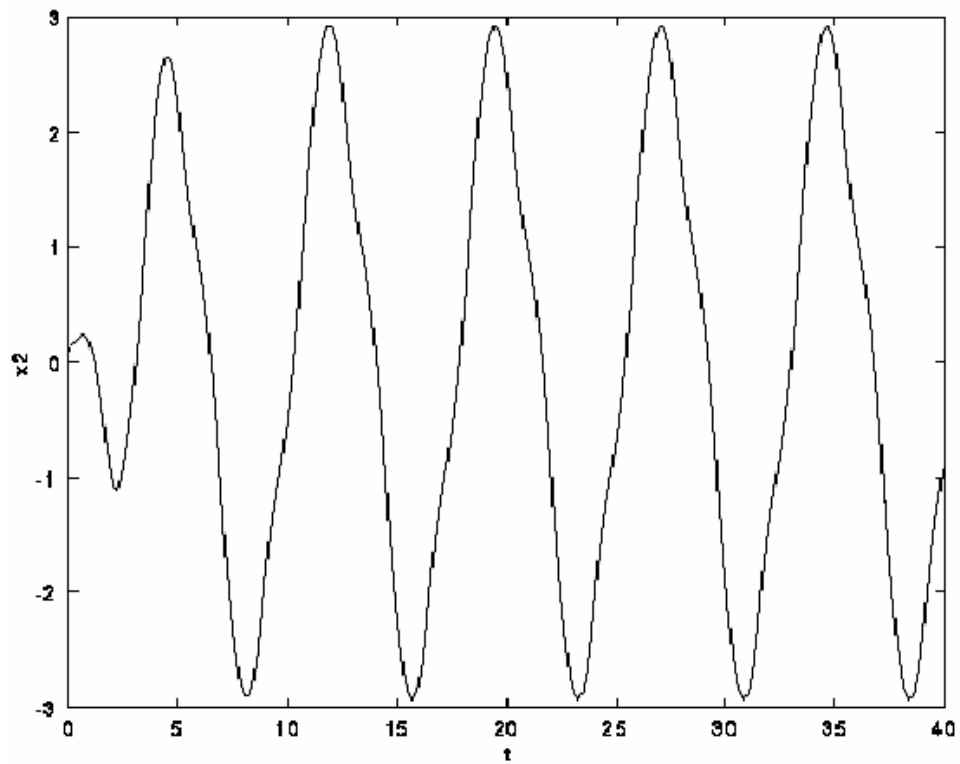
$$\begin{aligned} \dot{x}_1 &= -x_1 + \alpha y_1 - \beta y_2 \\ \dot{x}_2 &= -x_2 + \alpha y_2 - \beta y_1 \end{aligned} \quad (30)$$

Dạng sóng nghiệm của phương trình (30) tương ứng với $\alpha = 2, \beta = 2$ và điều kiện đầu $x_1(0) = 0.1, x_2(0) = 0.1$ được thể hiện trên hình vẽ 13a và 13b. Ta thấy rằng thay vì hội tụ đến một điểm cân bằng dc thì các biến trạng thái x_1 và x_2 lại hội tụ về dạng sóng tuần hoàn, tính tuần hoàn của CNN này được thể hiện rõ hơn qua đồ thị quỹ đạo trong mặt phẳng x_1 - x_2 như trong hình 13c. Mỗi điểm trên quỹ đạo xuất phát từ trạng thái đầu $(x_1, x_2) = (0.1, 0.1)$ ở thời điểm $t = 0$ đồ thị trong hình 13c sẽ được tham số hoá theo thời gian nhưng không thể hiện trên hình bởi vì chúng ta chỉ quan tâm đến quan hệ giữa $x_1(t)$ và $x_2(t)$ khi $t \rightarrow \infty$, quỹ đạo này gọi là

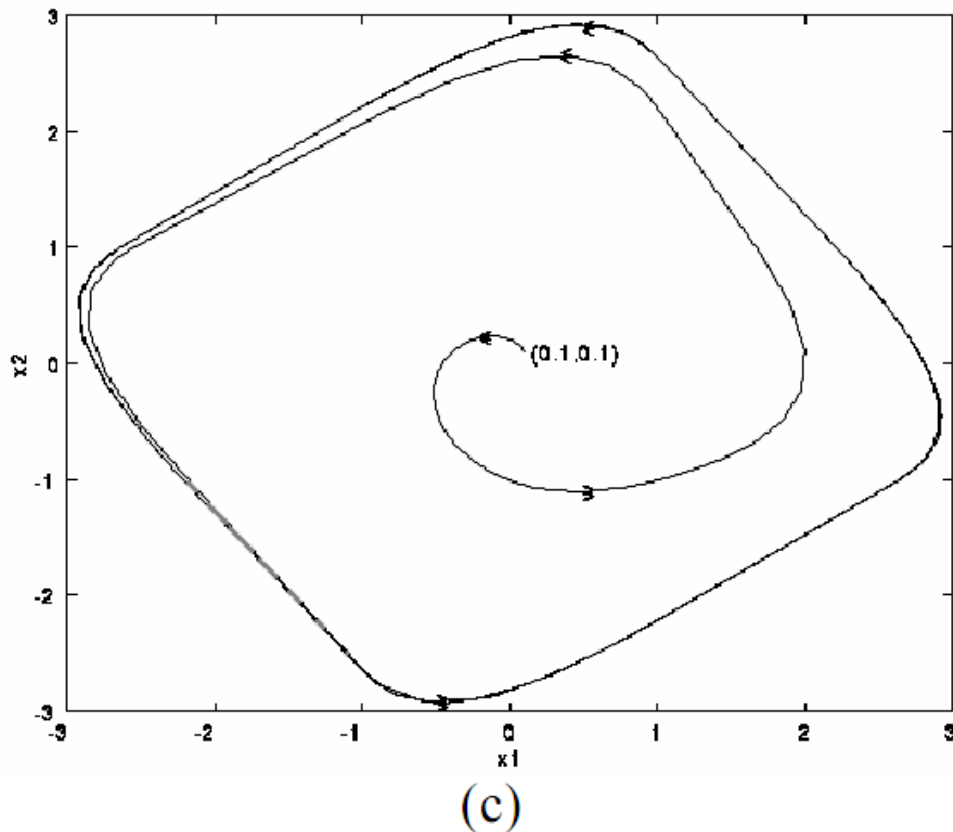
một *closed contour* hoặc một chu kỳ giới hạn. Do quỹ đạo từ $(0.1, 0.1)$ không hội tụ về một điểm cân bằng nên CNN này không là ổn định hoàn toàn.



(a)



(b)



Hình 13- Dạng sóng nghiệm tuần hoàn của $x_1(t)$ và $x_2(t)$ và quỹ đạo tương ứng trong trường hợp $\alpha = 2, \beta = 2, x_1(0) = 0.1, x_2(0) = 0.1$

Ví dụ CNN hỗn độn với 2 tế bào và 1 đầu vào dạng sin

Giả sử chúng ta áp dụng một đầu vào dạng sin: $u_{11}(t) = 4.04 \sin(\frac{\pi}{2}t)$ đối với tế bào C(11) của CNN 2 tế bào như hình 12 và chọn $\alpha = 2, \beta = 1.2$. Ở đây, cũng với điều kiện biên 0, ta có phương trình trạng thái các tế bào sau:

$$\dot{x}_1 = -x_1 + 2y_1 - 1.2y_2 + 4.04 \sin(\frac{\pi}{2}t) \quad (31)$$

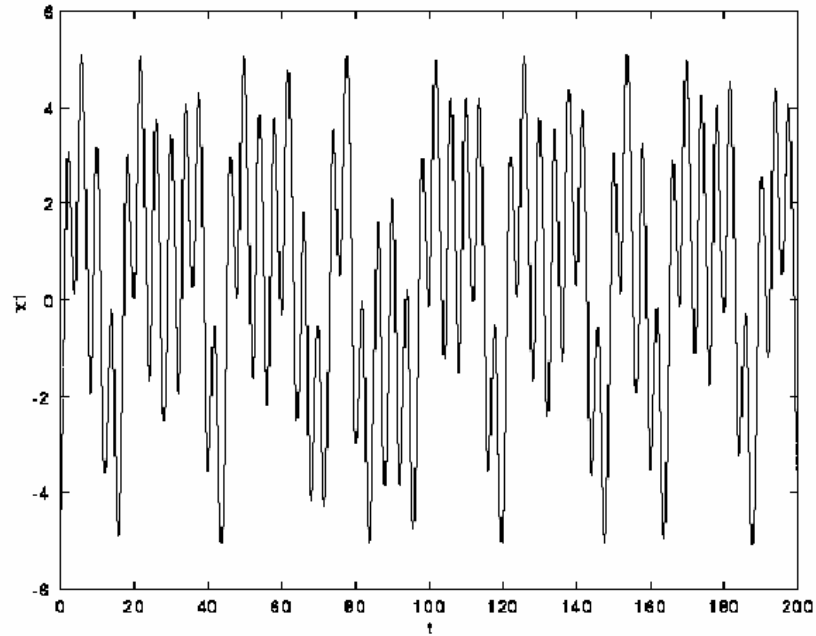
$$\dot{x}_2 = -x_2 + 1.2y_1 + 2y_2$$

Hệ phương trình này là phương trình trạng thái của CNN 1x2 với khuôn mẫu:

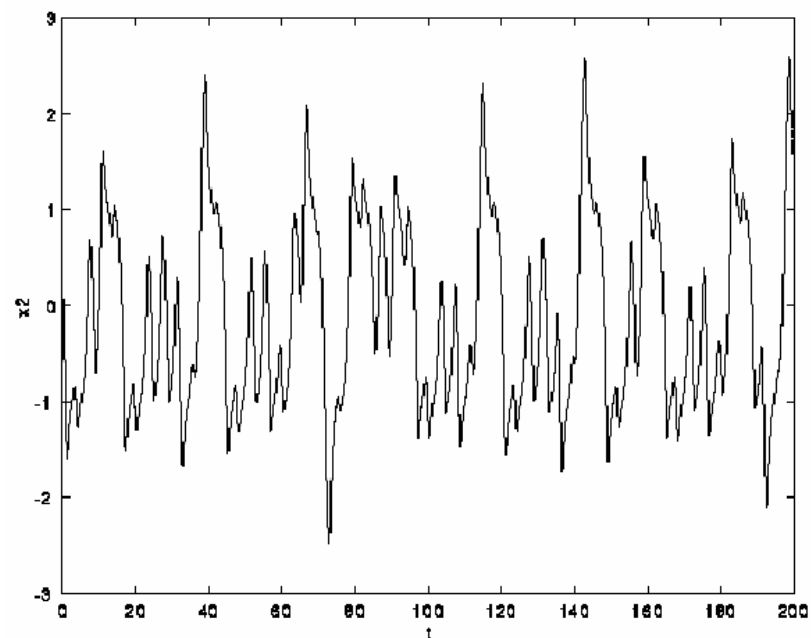
$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1.2 & 2 & -1.2 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0 \quad (32)$$

Với điều kiện biên 0, đầu vào dạng sin $u_{11}(t)$ đối với C(11) và đầu vào $u_{12} = 0$ đối với tế bào C(12). Dạng sóng nghiệm $x_1(t)$ và $x_2(t)$ tương ứng với điều kiện đầu $x_1(0) = 0.1$ và $x_2(0) = 0.1$ được thể hiện trên hình 14a và 14b. Ta thấy rằng khác với dạng nghiệm tuần hoàn trước đây, hai dạng nghiệm này không hội tụ tới một dạng tuần hoàn khi $t \rightarrow \infty$. Trên hình 14c ta thấy quỹ đạo giống như là cuộn chỉ rới không đầu. Hình 15a, 15b thể hiện phổ năng lượng

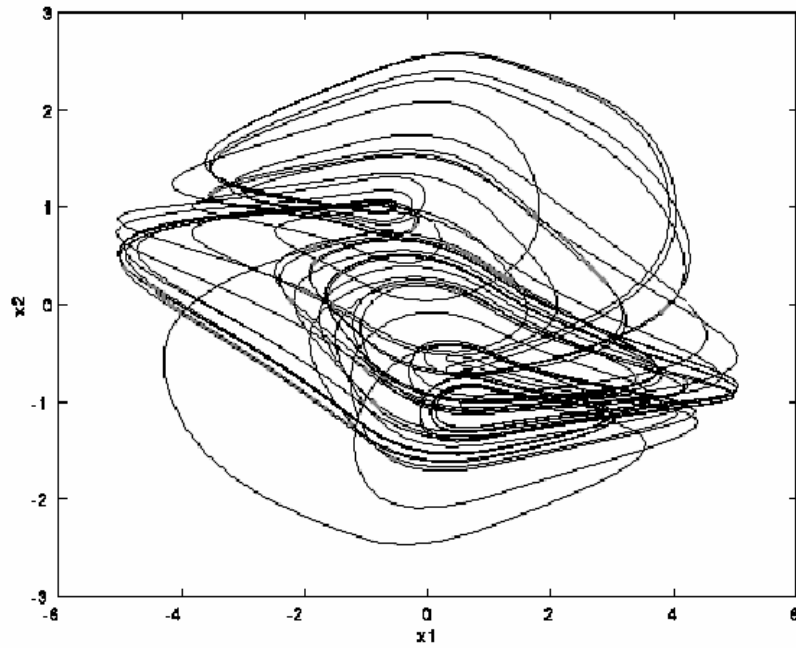
$X_1(\omega)$ của $x_1(t)$ và $X_2(\omega)$ của $x_2(t)$. Chúng có đặc trưng dạng nhiễu, liên tục, dải rộng khác với tín hiệu tuần hoàn chỉ chứa các đường rời rạc tương ứng với các thành phần điều hoà của khai triển Fourier của nó. Dạng nhiễu ở hình 14a, 14b được gọi là hỗn độn (*chaotic*), và quỹ đạo liên đới được gọi là quỹ đạo hấp dẫn lạ thường (*strange attractor*) do các nghiệm khác tương ứng với điều kiện đầu lân cận sẽ bị hút vào và hội tụ về cùng một quỹ đạo này.



(a)

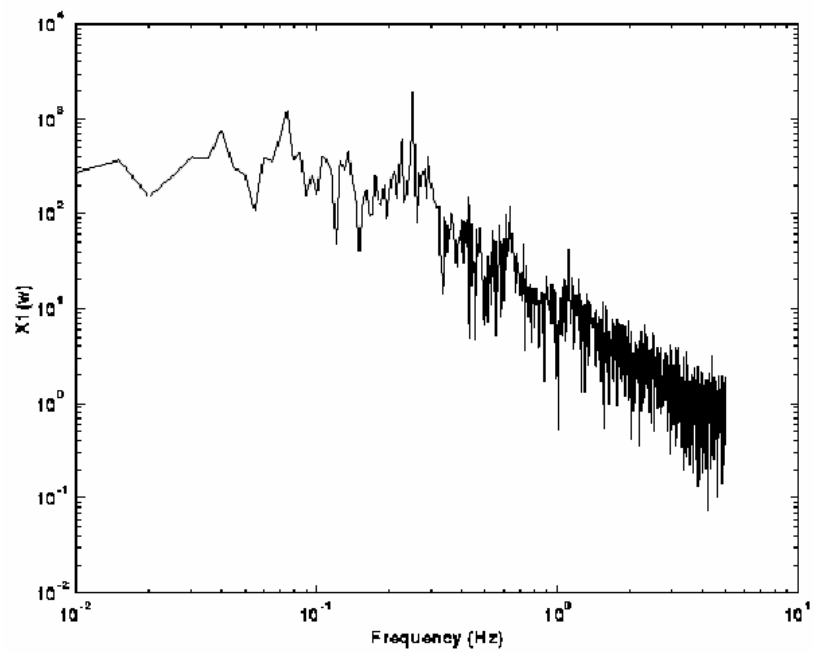


(b)

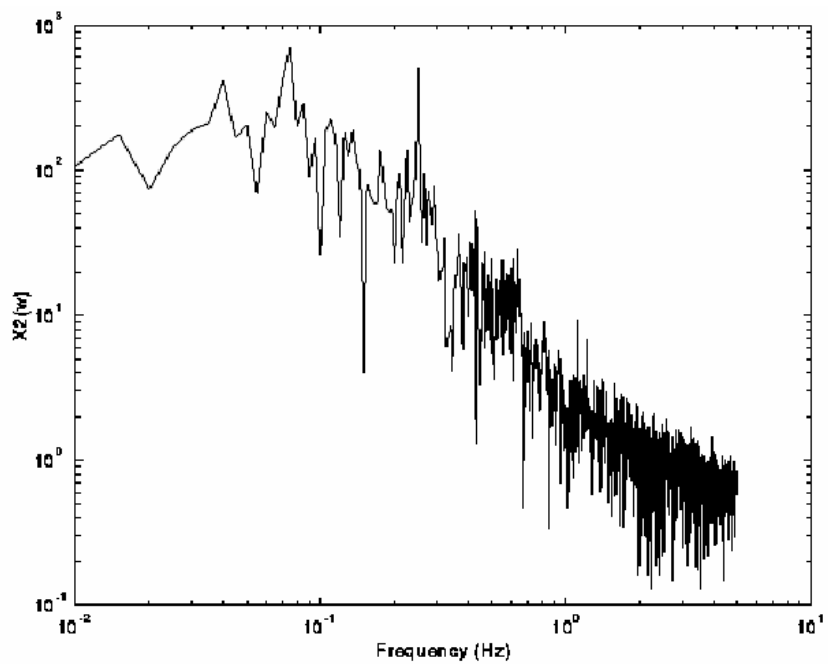


(c)

Hình 14 - Dạng nghiệm hỗn độn của $x_1(t)$ và $x_2(t)$ và quỹ đạo tương ứng với trường hợp $\alpha = 2, \beta = -1.2, x_1(0) = 0.1, x_2(0) = 0.1$



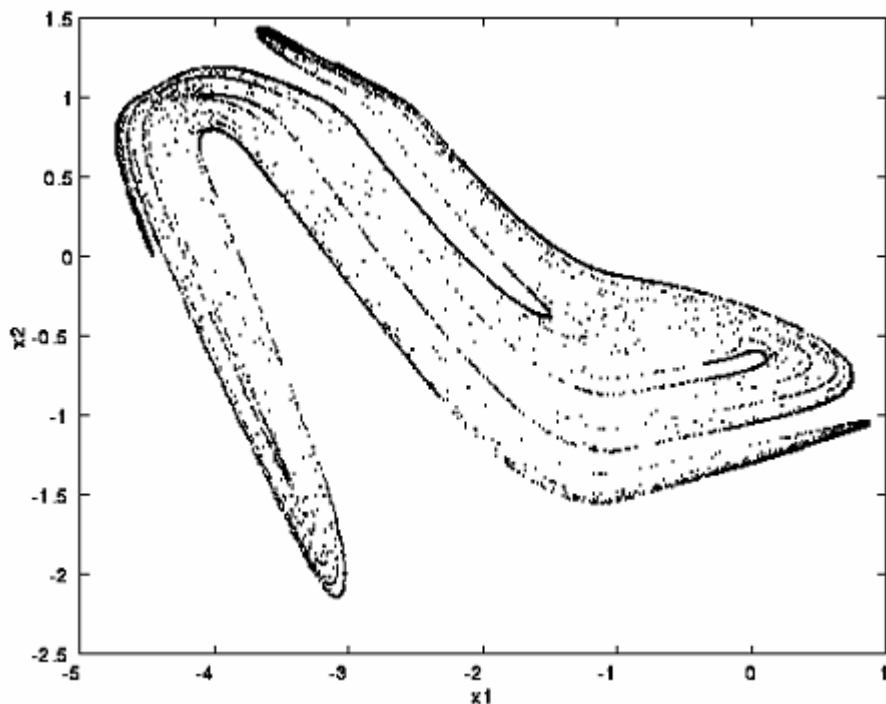
(a)



(b)

Hình 15- Phổ năng lượng tần số tính toán số học từ nghiệm hỗn loạn $x_1(t)$ và $x_2(t)$

Mặc dù quỹ đạo hấp dẫn lạ thường ở hình 14c trông cực kỳ lộn xộn, nó lại mang một cấu trúc hình học có trật tự nào đó. Trong trường hợp đầu vào tuần hoàn như ví dụ trên, nếu ta lấy mẫu từng điểm trên quỹ đạo trong mỗi chu kỳ của đầu vào thì ta được một tập hợp điểm gọi là *mặt cắt Poincare*, hoặc đơn giản là *bản đồ Poincare*. Trong ví dụ này, chu kỳ đầu vào \sin là $T=4$. Như vậy, nếu ta vẽ $(x_1(t), x_2(t))$ trên phẳng x_1-x_2 ở các thời điểm $t=0,4,8,12,16,\dots$ thì ta sẽ có được quỹ đạo hấp dẫn lạ thường như hình 16, thường được gọi là quỹ đạo hấp dẫn giày nữ.



Hình 16 - Bản đồ Poincare trích xuất từ quỹ đạo hấp dẫn lạ thường ở hình 13c được gọi là “hấp dẫn giày nữ”

Rẽ nhánh và hỗn độn trong CNN.

Ta hãy xem xét một CNN với 3 tế bào được mô tả bởi hệ phương trình trạng thái sau:

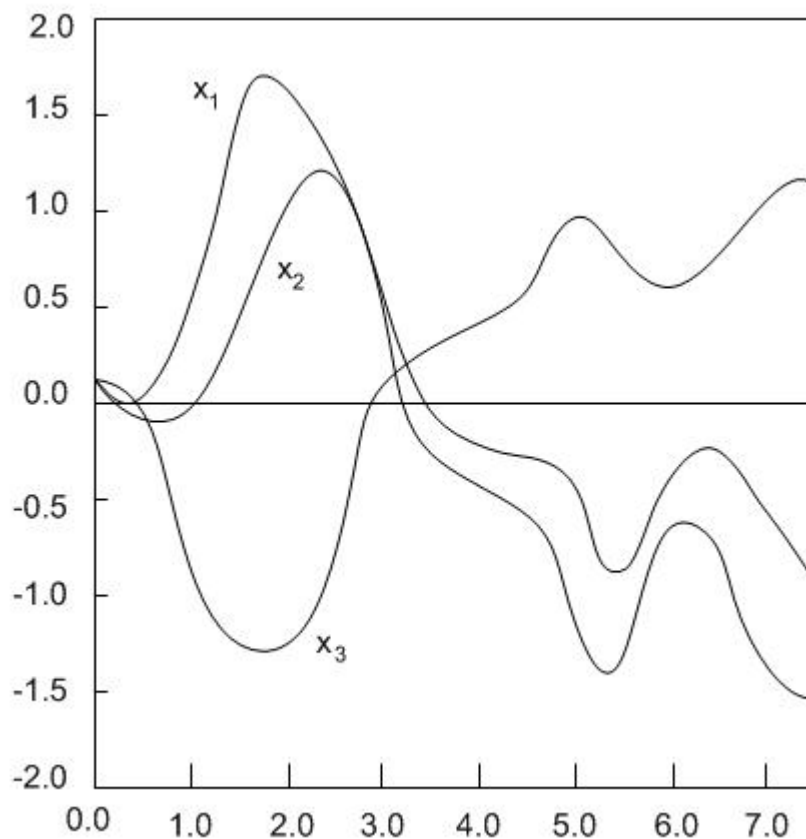
$$\begin{aligned}x_1' + x_1 &= p_1 f(x_1) - s f(x_2) - r f(x_3) \\x_2' + x_2 &= -s f(x_1) + p_2 f(x_2) - r f(x_3) \\x_3' + x_3 &= -s f(x_1) + r f(x_2) + p_3 f(x_3)\end{aligned}\quad (33)$$

Với $p_i > 1$, $p_2 > 1$, $p_3 > 1$, $r, s > 0$, đầu vào $u=0$ và độ dịch $I = 0$. Hàm đầu ra là được tính theo công thức

$$y_i = f(x_i) = \frac{1}{2}(|x_i + 1| - |x_i - 1|), i = 1, 2, 3. \quad (34)$$

Với CNN tự sinh (autonomous) 3 tế bào được mô tả bởi hệ động lực (33), (34) ta có:

- i) Với $|x_i| > 1$ và $p_i > 1$, $s, r > 0$, $i=1,2,3$ tất cả các điểm cân bằng của CNN là bền vững. Hơn thế, bất kỳ quỹ đạo nào trong vùng này đều tiệm cận về điểm cân bằng tương ứng.
- ii) Với $|x_i| < 1$ và $p_i > 1$, $i=1,2,3$ tất cả các điểm cân bằng của CNN đều không ổn định. Hình vẽ sau biểu diễn quỹ đạo hấp dẫn lạ thường của CNN này với các thông số $p_1=1,25$; $p_2=1,1$; $p_3=1$; $s=3,2$; $r=4,4$ và giá trị ban đầu $x(0)=(0,1;0,1;0,1)$



Hình 17: Quỹ đạo hấp dẫn lạ thường của CNN (3.3) với $p_1=1,25$; $p_2=1,1$; $p_3=1$; $s=3,2$; $r=4,4$ và giá trị ban đầu $x(0)=(0,1;0,1;0,1)$

Mạng nơ ron tế bào nhiều lớp

Thay vì chỉ có một biến trạng thái trong trường hợp lớp đơn thì ở mạng nơ ron tế bào nhiều lớp có nhiều biến trạng thái trong mỗi tế bào. Khái niệm nổi bật ở đây chính là các mối

tương tác của các biến trạng thái trong cùng 1 lớp. Để tránh sự hỗn loạn, ta sử dụng toán tử tích chập * để viết 2(a) sang dạng:

$$C \frac{dv_{xij}(t)}{dt} = \frac{-1}{R_x} v_{xij}(t) + A * v_{yij}(t) + B * v_{uij} + I, \\ 1 \leq i \leq M; 1 \leq j \leq N. \quad (35)$$

Với mạng nơ ron tế bào nhiều lớp, phương trình động lực của một tế bào có thể biểu diễn theo dạng sau:

$$C \frac{dv_{xij}(t)}{dt} = -R^{-1} v_{xij}(t) + A * v_{yij}(t) + B * v_{uij} + I, \\ 1 \leq i \leq M; 1 \leq j \leq N \quad (36)$$

trong đó:

$$C = \begin{pmatrix} C_1 & & 0 & 0 \\ & \ddots & & 0 \\ 0 & & & \\ 0 & 0 & & C_m \end{pmatrix} \quad R = \begin{pmatrix} R_{1x} & & 0 & 0 \\ & \ddots & & 0 \\ 0 & & & \\ 0 & 0 & & R_{mx} \end{pmatrix} \quad (37a)$$

$$A = \begin{pmatrix} A_{11} & 0 & 0 & 0 & 0 \\ \cdot & \cdot & 0 & & 0 \\ \cdot & & \cdot & 0 & 0 \\ \cdot & & & \cdot & 0 \\ A_{m1} & \cdot & \cdot & \cdot & A_{mm} \end{pmatrix} \\ B = \begin{pmatrix} B_{11} & 0 & 0 & 0 & 0 \\ \cdot & \cdot & 0 & & 0 \\ \cdot & & \cdot & 0 & 0 \\ \cdot & & & \cdot & 0 \\ B_{m1} & \cdot & \cdot & \cdot & B_{mm} \end{pmatrix} \quad (37b)$$

$$v_{xij} = \begin{pmatrix} v_{1xij} \\ \vdots \\ v_{mxij} \end{pmatrix} \quad v_{yij} = \begin{pmatrix} v_{1yij} \\ \vdots \\ v_{myij} \end{pmatrix} \quad v_{uij} = \begin{pmatrix} v_{1uij} \\ \vdots \\ v_{muij} \end{pmatrix} \quad (37c)$$

$$I = \begin{pmatrix} I_1 \\ \vdots \\ I_m \end{pmatrix} \quad (37d)$$

Ở đây m là số các biến trong mạch nhiều lớp tế bào.

Quan sát thấy rằng C và R là ma trận đường chéo, A và B là ma trận tam giác.

Nhận xét:

a) Với mạng nơ ron tế bào nhiều lớp, tất cả các kết quả giới thiệu ở trên vẫn đúng. Trạng thái ổn định có thể chứng minh từ lớp cuối cùng (lớp 1) tới lớp trên cùng bằng việc sử dụng cấu trúc tam giác của ma trận A và B.

b) Có nhiều biến trạng thái trong mạch tế bào, chúng ta có thể chọn nhiều đặc tính động đồng thời cho nhiều biến trạng thái khác nhau. Đặc điểm này làm cho mạng trở nên linh hoạt

hơn và cho phép chúng ta giải quyết với nhiều bài toán xử lý ảnh phức tạp hơn.

c) Sử dụng nhiều đặc tính động đề cập trong mục b), chúng ta có thể chọn nhiều hằng số thời gian khác nhau cho nhiều biến trạng thái khác nhau của mạch tế bào. Trong trường hợp tới hạn, chọn $C_q = 0$ cho biến trạng thái v_{qxij} , do đó sẽ đạt được bộ các phương trình đại số và phương trình vi phân. Đặc tính này sẽ giúp chúng ta thiết kế mạng nơ ron tế bào linh hoạt hơn cho các ứng dụng thực tế.

Mối quan hệ của CNN với phương trình vi phân đạo hàm riêng và ô tô mát tế bào

Nói chung mạng nơ ron tế bào có thể được mô tả bởi hệ thống của các phương trình vi phân thông thường. Tất cả các tế bào được sắp xếp trong mạng tuần tự, có tính đồng nhất. Có 2 mô hình toán học cũng có cấu trúc không gian đồng nhất có thể mô tả đặc tính động trong không gian. Một là hệ phương trình vi phân đạo hàm riêng, và thứ 2 là ô tô mát tế bào. Tuy nhiên đặc tính động của chúng phụ thuộc vào môi tương tác cục bộ trong không gian của chúng. Ta hãy xác định mối quan hệ giữa các mạng nơ ron tế bào với 2 mô hình toán học này.

Đầu tiên xét phương trình vi phân đạo hàm riêng. Phương trình nhiệt rất nổi tiếng từ vật lý là:

$$\frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} = \frac{1}{\kappa} \frac{du(x, y, t)}{dt} \quad (38)$$

Trong

đó κ là hằng số, gọi là suất dẫn nhiệt, lời giải $u(x, y, t)$ của phương trình nhiệt là hàm liên tục về thời gian t , và biến không gian x, y . Nếu ta xấp xỉ hàm $u(x, y, t)$ bằng tập các hàm $u_{ij}(t)$ được định nghĩa như sau:

$$u_{ij}(t) = u(ih_x, jh_y, t) \quad (39)$$

trong đó h_x , và h_y là các bước không gian trong tọa độ x, y , thì đạo hàm riêng của $u(x, y, t)$ với biến x, y có thể được tính xấp xỉ như sau:

$$\begin{aligned} \frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} \\ \approx \frac{1}{4} [u_{i,j-1}(t) + u_{i,j+1}(t) + u_{i-1,j}(t) + u_{i+1,j}(t)] \\ - u_{i,j}(t), \quad \text{for all } i, j. \end{aligned} \quad (40)$$

Phương trình nhiệt (38) lúc này có thể được xấp xỉ bởi tập các phương trình sau:

$$\begin{aligned} \frac{1}{\kappa} \frac{du_{ij}(t)}{dt} = \frac{1}{4} [u_{i,j-1}(t) + u_{i,j+1}(t) + u_{i-1,j}(t) + u_{i+1,j}(t)] \\ - u_{i,j}(t), \quad \text{for all } i, j. \end{aligned} \quad (41)$$

So sánh (41) và (2), chúng ta sẽ thấy nét tương tự nhau giữa hai phương trình. Cả hai phương trình là loại phương trình vi phân thông thường với các biến liên kết cục bộ mô tả các đặc tính động của hệ thống. Điểm khác biệt quan trọng giữa hai phương trình là phương trình tế bào (2) là phương trình vi phân *phi tuyến* thông thường, trong khi đó (41) là phương trình vi phân *tuyến tính* thông thường.

Đối với ô tô mát tế bào, hệ động lực được mô tả bằng phương trình sau:

$$a_{ij}(n+1) = \phi [a_{kl}(n) \text{ for all } C(k, l) \in N_r(i, j)]. \quad (42)$$

Nếu ta rời rạc hóa phương trình (2) với $B(i, j, k, l) = 0$ với mọi i, j, k, l , theo thời gian t thì ta có:

$$\frac{C}{h} [v_{xij}((n+1)h) - v_{xij}(nh)] = -\frac{1}{R_x} v_{xij}(nh) + \sum_{C(k,l) \in N_r(i,j)} A(i, j; k, l) v_{ykl}(nh) + I,$$

$$1 \leq i \leq M; 1 \leq j \leq N \quad (43a)$$

$$v_{yij}(nh) = 0.5R_y (|v_{xij}(nh) + 1| - |v_{xij}(nh) - 1|), \quad 1 \leq i \leq M; 1 \leq j \leq N. \quad (43b)$$

Sau khi sắp xếp lại (43a) và thế biểu thức kết quả cho $v_{xij}((n+1)h)$, ta có:

$$v_{yij}(n+1) = \phi' [v_{xij}(n), v_{ykl}(n) \text{ cho mọi } C(k, l) \in N_r(i, j)] \quad (44)$$

trong đó:

$$v_{yij}(n) \triangleq v_{yij}(nh), \quad v_{xij}(n) \triangleq v_{xij}(nh) \quad (45a)$$

$$\begin{aligned} & \phi' [v_{xij}(n), v_{ykl} \text{ cho mọi } C(k, l) \in N_r(i, j)] \\ &= g \left[\left(\frac{h}{CR_x} + 1 \right) v_{xij}(n) \right. \\ & \quad \left. + \frac{h}{C} \sum_{(k,l) \in N_r(i,j)} A(i, j; k, l) v_{ykl}(n) + \frac{hI}{C} \right] \end{aligned} \quad (45b)$$

và

$$g[z] = \frac{1}{2} (|z+1| - |z-1|). \quad (45c)$$

So sánh (42) và (44), một lần nữa chúng ta sẽ thấy điểm giống nhau đặc biệt giữa chúng. Điểm khác chính là với ô tô mát tế bào những biến trạng thái là giá trị nhị phân và hàm động lực là hàm logic của các trạng thái trước của những tế bào lân cận, trong khi mạng nơron tế bào có những biến trạng thái là giá trị thực, biến và hàm động lực là hàm thực phi tuyến của những trạng thái trước của những tế bào lân cận.

Bảng sau là phép so sánh CNN với ô tô mát tế bào và phương trình vi phân đạo hàm riêng là các hệ tính toán có tính cân đối trong không gian.

<i>Model</i>	<i>CNN</i>	<i>Partial differential equation</i>	<i>2D- cellular automata</i>
Thời gian	Liên tục	Liên tục	Rời rạc
Không gian	Rời rạc	Liên tục	Rời rạc
Trạng thái	Số thực	Số thực	Số nhị phân
Dynamics	Nonlinear	Linear/Nonlinear	Nonlinear

Kết luận

Ta đã khảo sát một cấu trúc mạch mới là mạng nơ ron tế bào có thể xử lý tín hiệu song song trong thời gian thực. Một số tính chất liên quan đến dải đặc tính động và trạng thái ổn định của mạng nơ ron tế bào đã được đề cập. Một số mô phỏng một vài đặc tính động của mạng nơ ron tế bào đơn giản trên máy tính điện tử đã cho các kết quả tương ứng với lý thuyết. Tính chất tương tác giữa tế bào với các tế bào lân cận gần nhất của mạng nơ ron tế bào có khả năng dễ dàng thực hiện bằng công nghệ VLSI hơn những mạng nơ ron khác.

Mặc dù mạng CNN chỉ có những tương tác cục bộ với các tế bào lân cận gần nhất, nhưng mạng CNN cũng có nhiều đặc điểm toàn cục bởi tính lan truyền của mạng. Đặc tính lan truyền qua các trọng số liên kết A, B này sẽ cho chúng ta khả năng lái mạng theo quỹ đạo tới các trạng thái mong muốn trong các bài toán nhận dạng và xử lý ảnh nhanh. ***Khả năng điều khiển các trọng liên kết cục bộ của tế bào sẽ làm thay đổi đặc tính động của toàn mạng là điểm nổi bật, đáng lưu ý nhất trong hoạt động của mạng nơ ron tế bào CNN.***

Xa hơn nữa, mạng nơ ron tế bào có dải động thực tế, mà các mạng nơ ron thông thường khó thực hiện được trên mạch điện.

1.3. Máy tính vạn năng mạng nơ-ron tế bào CNN UM

a. Sự phát triển của máy tính điện tử

Công cụ tính toán đã được loài người phát minh ra trên 6000 năm từ khi còn dùng các ngón tay, viên sỏi, bàn tính vv... để tính toán. Tuy nhiên lịch sử phát triển của máy tính điện tử chạy theo chương trình thì chỉ mới 60 năm nay.

Tiền thân của máy tính điện tử là máy tính sử dụng rơ le đầu tiên do giáo sư Harvard Aiken ở đại học Harvard thiết kế và hãng IBM chế tạo năm 1941 mang tên Mark I. Máy gồm 760 000 rơ le, 800 km dây nối và thực hiện được phép cộng hai số trong 1/3 giây. Tốc độ này quá chậm do linh kiện chuyển mạch trạng thái là rơ le có độ trễ lớn nên xuất hiện nhu cầu phát triển linh kiện chuyển mạch nhanh hơn, và trên cơ sở linh kiện chuyển mạch bắt đầu hình thành các máy tính điện tử ở các thế hệ khác nhau.

Máy tính sử dụng bóng đèn điện tử đầu tiên là máy EINAC gồm 18000 bóng đèn điện tử, 6000 công tắc và 1500 rơ le có khả năng thực hiện 5000 phép cộng trong một giây, công suất tiêu thụ 140 kW, có kích thước dài 30m, rộng 1m, cao 3m và trọng lượng nặng tới 30 tấn. Việc lập trình cho máy tính này thông qua việc kết nối các dây dẫn tương tự như các tổng đài điện thoại lúc đó. Neumann János người Hungary năm 1944 tình cờ gặp kỹ sư trưởng của máy tính EINAC là Goldstine và đã được Goldstine giới thiệu về hoạt động của máy EINAC. Sau khi xem xét kỹ máy, Neumann János đã phát hiện ra các chương trình máy tính cũng có thể lưu trong máy như các số liệu. Như vậy ta có thể dạy cho máy biết phân biệt đâu là lệnh đâu là dữ liệu và lập trình cho máy chạy theo chương trình lưu trong bộ nhớ chứ không phải bằng cách nối dây. Năm 1946 Neumann, Goldstine và Burks đã công bố phát minh về cấu trúc của máy tính điện tử có điều khiển theo chương trình này. Từ đó đến nay các máy tính điện tử đều hoạt động theo nguyên lý này trên cơ sở hệ nhị phân tương thích với hai trạng thái đóng- mở của linh kiện chuyển mạch.

Năm 1947 ba nhà khoa học Mỹ là W. H. Brattain, J. Bardeen và W. Shockley đã phát minh ra bóng bán dẫn tại phòng thí nghiệm Bell. Máy tính điện tử thế hệ 2 được chế tạo bằng bóng bán dẫn với bộ nhớ xuyên ferrite đã có kích thước nhỏ hơn nhiều lần so với thế hệ thứ nhất dùng bóng đèn điện tử. Sự phát triển tiếp là các vi mạch tích hợp IC (Integrated Circuit) với độ tích hợp số bóng bán dẫn trong chip ngày càng cao vào cuối thập kỷ 1960 cho ra đời các máy tính thế hệ 3 là các máy mainframe và mini-computers. Đến năm 1971 bộ vi xử lý đầu tiên đã được Intel chế tạo mở đầu cho các máy tính cá nhân IBM PC, Sinclair, Commodore ra đời. Mật độ tích hợp giai đoạn này lên đến hàng trăm ngàn bóng bán dẫn trên 1cm^2 .

Máy tính điện tử thế hệ 4, 5 đã có nhiều bộ vi xử lý và chạy được nhiều chương trình song song. Độ phức tạp của các chip ngày càng tăng và tốc độ tính toán ngày càng cao đã đưa các máy tính điện tử truyền thống đến giới hạn vật lý về kích thước và tốc độ xử lý.

Các máy tính số hiện nay về cơ bản vẫn là loại máy logic với các dữ liệu rời rạc được mã hóa theo hệ nhị phân. Tính chất chung của nó là khả năng thực hiện thuật toán theo chương trình được lưu trong bộ nhớ. Đây là loại máy tính vạn năng xử lý trên các số nguyên (Universal Machine on Integers) hay còn gọi là máy Turing (Turing Machine). Các phép tính cơ bản của nó là các phép số học và logic. Thuật toán (algorithms) là các chuỗi logic của các phép tính cơ bản này. Từ năm 1960 đến năm 2000 độ tích hợp của chip IC tăng từ 1 bóng bán dẫn đến con số hàng trăm triệu bóng bán dẫn/chip. Tuy nhiên về cấu trúc và nguyên lý hoạt động của các vi xử lý này không có gì khác so với nguyên lý của Neumann János đưa ra từ năm 1946.

Sự ra đời của các vi mạch tích hợp rất lớn VLSI đã tạo ra các máy tính số có tính thực tiễn cao với giá thành rẻ và hiện nay đã trở thành một loại hàng hóa thông dụng.

Mặc dù vậy các máy tính hiện đại ngày nay còn gặp nhiều khó khăn trong một số bài toán mà các sinh vật sống xử lý rất đơn giản như kiểm soát đi lại, ăn uống và tìm môi trường... Do vậy cần phải có một nguyên lý tính toán mới, cấu trúc mới để tiếp tục nâng cao được khả

năng tính toán và giải quyết được những vấn đề mà máy tính điện tử hiện hành chưa giải quyết được.

Sự ra đời của mạng nơ ron tế bào CNN đã mở ra một hướng mới cho sự phát triển của khoa học tính toán tiếp cận đến các phương thức xử lý cũng như phương thức cảm nhận và hành động của các tổ chức trong cơ thể sinh vật sống.

Năm 1993 Giáo sư Roska Tamás ở Viện Nghiên cứu Máy tính và Tự động hóa Hungary và Giáo sư L. O. Chua ở đại học Berkeley Mỹ đã công bố nguyên lý máy tính CNN mới này tại Viện Hàn lâm khoa học Hungary. Không lâu sau giáo sư Angel Rodriguer Vazquez ở Seville Tây Ban Nha cùng hợp tác với nhóm CNN Budapest-Berkeley và cho ra đời Chip CNN CP400 đầu tiên năm 1995, đánh dấu một hướng phát triển mới của máy tính điện tử [08].

Với chip mạng nơ ron tế bào CNN-UM chương trình vẫn được lưu trong bộ nhớ nhưng các phép tính đã được thực hiện song song trong môi trường tín hiệu tương tự. Năm 1999 chip CNN với đầu vào quang học đầu tiên đã ra đời với 4096 CPU có khả năng xử lý đến 50 000 ảnh/giây. Tốc độ xử lý này tương đương với 9200 bộ vi xử lý Pentium. Chip CNN 256x256 CPU đang được thiết kế có tới 64000 CPU, có khả năng xử lý các chức năng mà hiện nay chỉ được ứng dụng trong các máy bay quân sự hoặc trong các hệ thống xử lý dữ liệu tài chính quốc gia.

Việc lập trình cho các chip CNN được thực hiện qua các ma trận trọng kết nối của mạng nơ ron tế bào (A, B, z). Các ma trận này được thực hiện đồng thời trên toàn mạng tạo nên một máy tính có hệ động lực xử lý tín hiệu hỗn hợp tương tự - số trong cả miền không gian và thời gian.

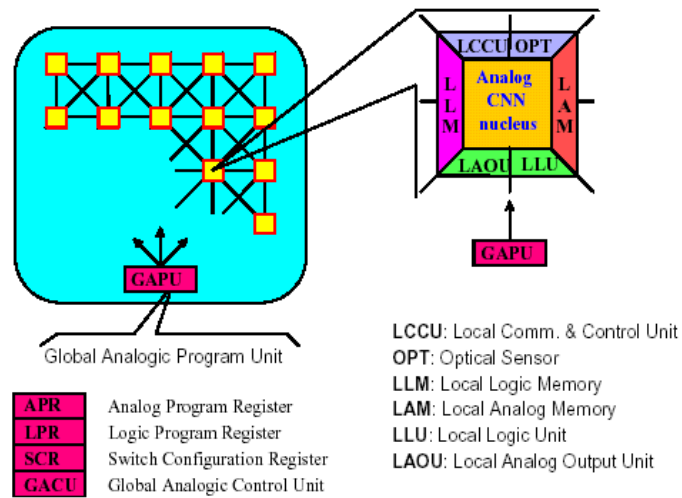
Để dễ dàng lập trình cho chip CNN cần có các công cụ phù hợp như ngôn ngữ lập trình bậc cao, hệ điều hành. Các công cụ này đã được Viện MTASzTAKI của Hungary phát triển và tạo nền tảng cho các máy tính CNN hoạt động ở Châu Âu, Mỹ và Nhật Bản. Với các công cụ này máy tính CNN có tốc độ tính toán tới Tera OPS gấp hàng trăm lần tốc độ xử lý của các máy tính hiện hành.

b. Máy tính vạn năng mạng nơ ron tế bào CNN-UM (*Cellular Neural Network Universal Machine*) [26][31][08]

Cấu trúc phần cứng

Máy tính vạn năng CNN-UM có thể được coi là mảng các tế bào xử lý phi tuyến đa chiều được liên kết cục bộ. CNN-UM có khả năng xử lý tín hiệu hỗn hợp tương tự số do vậy có tên Analogic (Analog-logic) computer. Khác với các máy tính lai (hybrid), ở máy tính CNN không có các bộ biến đổi A/D và D/A và cũng không có khái niệm biểu diễn các giá trị tương tự bằng số. Tất cả tín hiệu và phép tính cơ bản đều là tương tự hoặc logic.

Cấu trúc của máy CNN-UM được mô tả trong Hình 18. Mạng nơ ron tế bào gồm các tế bào có cấu trúc đồng nhất. Lõi của các tế bào này thực hiện chức năng của hệ động lực chuẩn tế bào CNN như mô tả trong hình 2 (Analog CNN nucleus). Để mạng có thể hoạt động và có khả năng lập trình thay đổi các trọng liên kết, ở mỗi tế bào đều có bộ nhớ, các thanh ghi, các khối vào ra và truyền thông cục bộ. Ngoài ra toàn mạng còn có khối lập trình toàn cục GAPU (Global Analogic Program Unit) phục vụ cho điều hành toàn mạng.



Hình 18: Cấu trúc máy tính CNN-UM

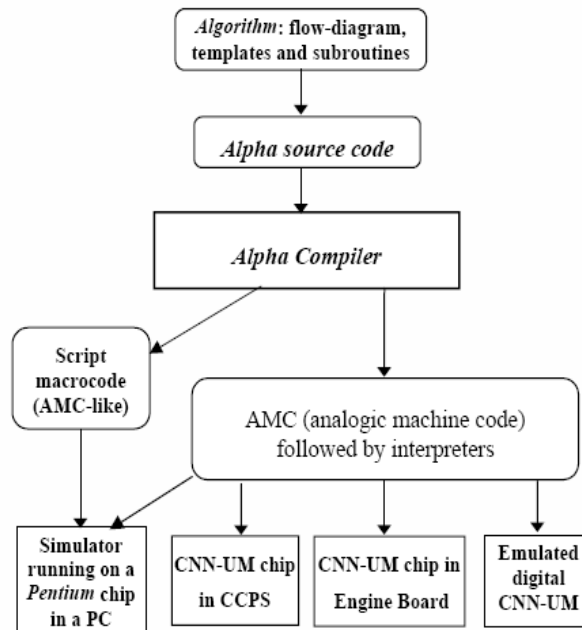
Khối OPT (Optical Sensor) ở mỗi tế bào làm chức năng thu nhận tín hiệu (điểm ảnh) đầu vào trực tiếp cho tế bào mà không cần bộ chuyển đổi A/D. Bộ nhớ cục bộ analog (LAM) và logic (LLM) lưu trữ các giá trị analog và logic của tế bào. Khối xử lý logic cục bộ (LLU) và khối xử lý đầu ra tương tự cục bộ (LAOU) thực hiện các phép tính toán logic và tương tự cho mỗi tế bào của mình. Các kết quả của mỗi tế bào được lưu giữ trong các bộ nhớ cục bộ. Khối điều khiển và truyền thông cục bộ (LCCU) thực hiện chức năng điều khiển và trao đổi thông tin giữa các tế bào lân cận và tới khối lập trình toàn cục (GAPU). Khối lập trình tương tự - số toàn cục GAPU có các thanh ghi và khối điều khiển toàn cục. Thanh ghi chương trình analog APR lưu trữ các trọng số của tế bào mạng CNN. Trong trường hợp $r = 1$ tổng số trọng lưu trữ là 19 số thực. Thanh ghi chương trình logic (LPR) chứa các lệnh logic cần thực hiện cho các tế bào. Thanh ghi cấu hình chuyển mạch (SCR) chứa các thông số khởi động và các tham số cho các chức năng hoạt động của các tế bào. Khối điều khiển tương tự số toàn cục (GACU) lưu các lệnh tuần tự của chương trình chính (analogic) và thực hiện các chức năng điều khiển toàn cục khác. Một thuật toán chạy trong CNN-UM có thể được tuần tự thực hiện qua các lệnh analog và logic. Mức độ kết quả trung gian có thể được tổ hợp và lưu trữ cục bộ. Các phép tính analog (*analog operations*) được định nghĩa bởi các trọng liên kết tuyến tính hoặc phi tuyến. Đầu ra có thể được định nghĩa cả trong trạng thái xác định hoặc không xác định của mạng. Các phép tính logic (NOT, AND, OR, ...) và số học (cộng, trừ) có thể được thực hiện trong mỗi tế bào. Dữ liệu có thể chuyển đổi được giữa các bộ nhớ cục bộ LAM và LLM.

Ngôn ngữ, chương trình dịch, hệ điều hành của CNN-UM

Cũng như các máy tính điện tử khác CNN-UM có các ngôn ngữ để lập trình từ mức thấp đến cao. Ở mức thấp nhất là mã máy, tiếp đến là ngôn ngữ assembly của CNN được gọi là AMC (Analogic Macro Code). Mã AMC được dịch thành mã máy dưới dạng firmwave và các tín hiệu điện cho chip CNN hoạt động. Ở mức cao có ngôn ngữ α mô tả các chu trình xử lý, mẫu trọng số, các chương trình con. Chương trình dịch α sẽ chuyển các lệnh ngôn ngữ α

sang dạng hợp ngữ AMC để chạy trên máy CNN. AMC có thể cho chạy trên phần cứng có chip CNN. Hệ điều hành COS (CNN Operating System) được cài đặt trên các máy CNN-UM phục vụ cho chạy các chương trình AMC cũng như giao tiếp với các hệ thống kết nối bên ngoài. Để phục vụ nghiên cứu và đào tạo ta có thể cho chạy chương trình AMC trong môi trường mô phỏng số CNN-UM (Emulated Digital CNN-UM) hoặc mô phỏng mềm (CNN simulator) trên máy PC Pentium với hệ điều hành Windows hoặc Unix.

Hình 19 mô tả khái quát các bước lập trình ở các ngôn ngữ khác nhau cho máy tính CNN-UM.



Hình 19: Các mức lập trình cho máy CNN-UM

Định nghĩa toán học của máy tính vạn năng mạng nơron tế bào - Máy tính sóng (Wave Computer)

Ta hãy khảo sát kỹ hơn về mặt toán học định nghĩa thế nào là máy tính vạn năng CNN. Máy tính CNN thực chất là máy tính xử lý dòng dữ liệu mảng như chuỗi ảnh video, mảng vectơ áp lực ở các tế bào xúc giác v.v.... Để định nghĩa một hệ tính toán ta cần định nghĩa *Dữ liệu (Data)*, *Lệnh (Instructions)* và *Thuật toán (Algorithm)* của nó. Đối với máy tính vạn năng CNN-UM ta có các định nghĩa đó như sau:

Dữ liệu(Data):

Chuỗi dữ liệu 2D (ví dụ ảnh video) $\phi(t)$ được định nghĩa như sau

$$\phi(t) : \{\varphi_{ij}(t), t \in T = [0, t_d]\} \quad (46)$$

$$1 \leq i \leq m; 1 \leq j \leq n$$

Ở đây m và n là các số nguyên, $t_d > 0$ là khoảng thời gian khảo sát, $\varphi_{ij}(t) \in C^1$ (là hàm liên tục, khả vi và bị chặn). φ_{ij} có thể là biến vào, biến trạng thái hay biến đầu ra của tế bào (1 pixel) trong mảng $m \times n$ tế bào. Ở thời điểm $t = t'$ ta có $\phi(t')$ là một ảnh ($n \times m$) pixel

$$P: \{p_{ij} \in \mathbb{R}^1\}, |p_{ij}| \leq P_{\max} \in \mathbb{R}^1 < \infty; p_{ij} \text{ là cường độ pixel} \quad (47)$$

Mức xám của ảnh được mô tả trong vùng +1 và -1 (+1 là trắng và -1 là đen). Ảnh màu được mô tả bằng tổ hợp của nhiều lớp ảnh $m \times n$ pixel, mỗi lớp mô tả cường độ của một màu (ví dụ hệ màu R.G.B).

Ảnh nhị phân được gọi là mặt nạ M

$$M: m_{ij} \in [1, -1] \quad (48)$$

Chuỗi ảnh ở các thời điểm $t_0, t_0 + \Delta t, t_0 + 2\Delta t \dots$ được gọi là dòng ảnh hay dòng video.

Lệnh (Instructions) :

Lệnh cơ bản của máy tính CNN được định nghĩa là:

$$\phi_{\text{output}}(t) = \psi\{\phi_{\text{input}}(t)\}, t \in T = [0, t_d] \quad (49)$$

Ở đây ψ là hàm của dòng ảnh. Ví dụ ta có thể chuyển đổi một video clip sang một video clip khác. Ta định nghĩa một phiên hàm F biến đổi một dòng ảnh sang một ảnh như sau:

$$P := F(\phi_{\text{input}}(t)) \quad (50)$$

Như vậy máy tính CNN có các dữ liệu ban đầu là dòng video, ảnh và mặt nạ: $\phi(0), P, M$. Lệnh cơ bản của máy CNN là phép giải phương trình vi phân đạo hàm riêng ψ trên dòng mảng dữ liệu $\phi(t)$.

Thuật toán (algorithms) :

Thuật toán của máy CNN chính là tổ hợp số học và logic của các dữ liệu và lệnh trên dòng mảng dữ liệu (ảnh) này. Ta gọi các thuật toán chạy trên các dòng mảng dữ liệu ở máy CNN là hàm đệ quy loại α (α -recursive function), trong khi ở máy tính số Von Neumann các thuật toán chạy trên các số nguyên là các hàm đệ quy loại μ (μ -recursive function).

Khả năng và độ phức tạp tính toán của máy CNN-UM [27]

Ở trên ta đã đề cập đến lệnh cơ bản của máy tính CNN là lời giải của các phương trình vi phân đạo hàm riêng PDE phi tuyến loại phản ứng–khuyếch tán (nonlinear reaction–diffusion equation). Ta hãy đi sâu hơn để hiểu rõ nguyên lý này. Phương trình đạo hàm riêng loại phản ứng – khuyếch tán được mô tả về mặt toán học có dạng:

$$\frac{\partial \phi(x, y, t)}{\partial t} - \text{div}(c(\phi(x, y, t)) \text{grad} \phi(x, y, t)) = \phi_0(x, y, t_0) + \varphi(\phi(x, y, t)) \quad (51)$$

Phương trình trên mô tả một loạt lớp phương trình PDE bao gồm

- Phương trình khuyếch tán tuyến tính khi $\phi_0 = 0$ và $\varphi(\cdot) = 0$
- Phương trình khuyếch tán tuyến tính bị chặn khi $\phi_0 \neq 0$ và $\varphi(\cdot) = 0$
- Phương trình sóng trigơ phi tuyến khi $\phi_0 = 0$ và $\varphi(\cdot) \neq 0$
- Phương trình sóng trigơ phi tuyến có giới hạn khi $\phi_0 \neq 0$ và $\varphi(\cdot) \neq 0$

Khi rời rạc hóa theo không gian, phương trình đạo hàm riêng trên trở thành một hệ phương trình vi phân thường liên kết cục bộ có dạng

$$\frac{d\phi_{ij}(t)}{dt} = g(\phi_{ij}(t)) - \phi_{ij}(t) + \frac{c_1}{4} (\phi_{i-1,j}(t) + \phi_{i+1,j}(t) + \phi_{i,j-1}(t) + \phi_{i,j+1}(t)) + z_{ij} \quad (52)$$

$$\phi_{ij}(t) = f(x_{ij}(t))$$

$$g(\cdot) = C_0 f(\cdot)$$

$$z_{ij} = z_0 + \sum_{kl \in N_1} B_{kl} \phi_{kl}(t_0)$$

Tương tự như dạng liên tục ta có các dạng PDE rời rạc sau:

- Phương trình khuếch tán tuyến tính khi

$$z_{ij} = 0 \text{ và } f(\phi) = \phi$$

- Phương trình khuếch tán tuyến tính bị chặn khi

$$z_{ij} \neq 0 \text{ và } f(\phi) = \phi$$

- Phương trình sóng trigơ phi tuyến khi

$$z_{ij} = 0 \text{ và } f(\phi) = \text{sign}(\phi)$$

- Phương trình sóng trigơ phi tuyến bị chặn khi

$$z_{ij} \neq 0 \text{ và } f(\phi) = \text{sign}(\phi)$$

Tất cả các dạng phương trình PDE rời rạc trên đều có thể lập trình trên mạng CNN sử dụng các mẫu liên kế

$$A = \begin{bmatrix} 0 & c_1 & 0 \\ c_1 & c_0 & c_1 \\ 0 & c_1 & 0 \end{bmatrix}; \quad B = \begin{bmatrix} b_2 & b_1 & b_2 \\ b_1 & b_0 & b_1 \\ b_2 & b_1 & b_2 \end{bmatrix}; \quad z_0 \quad (53)$$

Với $c_0 = 0$ và $c_1 > 0$ ta có lời giải phương trình vi phân đạo hàm riêng khuếch tán và với $c_0 > c_1 > 0$ ta có lời giải của phương trình sóng trigơ.

Như vậy bằng một lệnh $[A, B, z]$ ta có thể có lời giải cho phương trình vi phân đạo hàm riêng phi tuyến dạng phản ứng – khuếch tán trong thời gian $5\mu s$ là thời gian quá độ của mạch điện tử của 1 tế bào trong mạng CNN. Với tính chất giải các phương trình sóng trong một lệnh, ta còn gọi các máy tính vạn năng CNN là các máy tính sóng (Wave Computer) [15] [36]. Mặc dù vậy máy CNN cũng có tính vạn năng như các máy Turing [20].

Để xác định khả năng tính toán của máy tính ta cần có các số đo cụ thể như tốc độ, công suất tiêu thụ và diện tích (hoặc thể tích) của chip xử lý. Đối với chip CNN ACE16K ta có tốc độ tính toán lên tới 12 Tera OPS. Tốc độ tính toán của máy CNN phụ thuộc chủ yếu vào các tham số của bài toán [32]. Độ phức tạp của tính toán phụ thuộc chủ yếu vào các tham số của các trọng liên kết và không phụ thuộc vào kích cỡ của mạng. Điều này trái ngược với các máy tính số hoạt động theo chế độ tuần tự hiện nay.

Điểm đặc thù trong độ phức tạp tính toán ở máy tính CNN là tính chất liên tục trong thời gian và trong giá trị. Với tính chất này chênh lệch về độ phức tạp của tính toán giữa máy Turing và máy CNN là rất lớn có thể lên đến 1: 8000 lần khi xác định biên của các ảnh điện tim có nhiễu.

Sự khác nhau giữa máy tính số Turing và máy tính vạn năng CNN được tóm tắt trong bảng 1.

	Máy tính Turing UMZ (Universal Machine Over Z)	Máy tính CNN UMF (Universal Machine Over Flow)
Trường I/O	Số nguyên Z	F (dòng ảnh $R^{m \times n}$)
Lệnh cơ bản	Logic	Vi phân đạo hàm riêng
Phương thức hoạt động	Lặp	Bán lặp
Vùng tác động của lệnh cơ bản	Cục bộ	Toàn cục
Kiến trúc	Máy Turing	CNN vạn năng
Mô hình tính toán	- Cú pháp - Hàm đệ quy từng phần trên số nguyên	- Phương trình vi phân 2D, 3D - Hàm đệ quy α trên dòng ảnh

Bảng 1.3-1: So sánh nguyên lý hoạt động của máy tính số và máy tính vạn năng CNN

Tóm lại mạng nơ-ron tế bào đã mở ra một hướng mới cho sự phát triển của khoa học tính toán. Đây là một lĩnh vực khoa học công nghệ mới mẻ đầy triển vọng cho đa dạng ứng dụng. Nghiên cứu về mạng nơ-ron tế bào cho ta một khả năng khám phá ra các cơ chế hoạt động của các tổ chức trong cơ thể con người và các quy luật sự sống khác. Với sự phát triển của công nghệ, các ứng dụng của CNN sẽ giải quyết nhiều vấn đề mà các hệ máy tính hiện hành chưa giải quyết được [19]. Với các tính năng vượt trội, một ngày không xa công nghệ CNN thâm nhập vào các hệ nhúng, hệ cơ điện tử tạo nên các sản phẩm và hệ thống thông minh có những chức năng xử lý tương tự như con người.

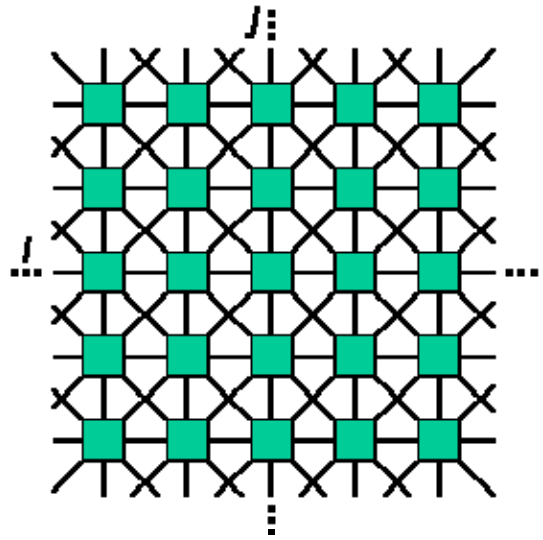
Để theo kịp xu thế phát triển của công nghệ thông tin, Việt nam cũng cần sớm đưa chương trình đào tạo, mở các môn học mới về tính toán nơ-ron, công nghệ mạng nơ-ron tế bào vào các trường đại học cho các bậc đào tạo đại học và sau đại học. Đặc biệt cần chú trọng các nghiên cứu cơ bản về mạng nơ-ron tế bào trong giai đoạn hiện nay.

1.4. Công nghệ xử lý ảnh nhanh trên nền mạng CNN

1.4.1. Máy tính xử lý ảnh nhanh CNN [7]

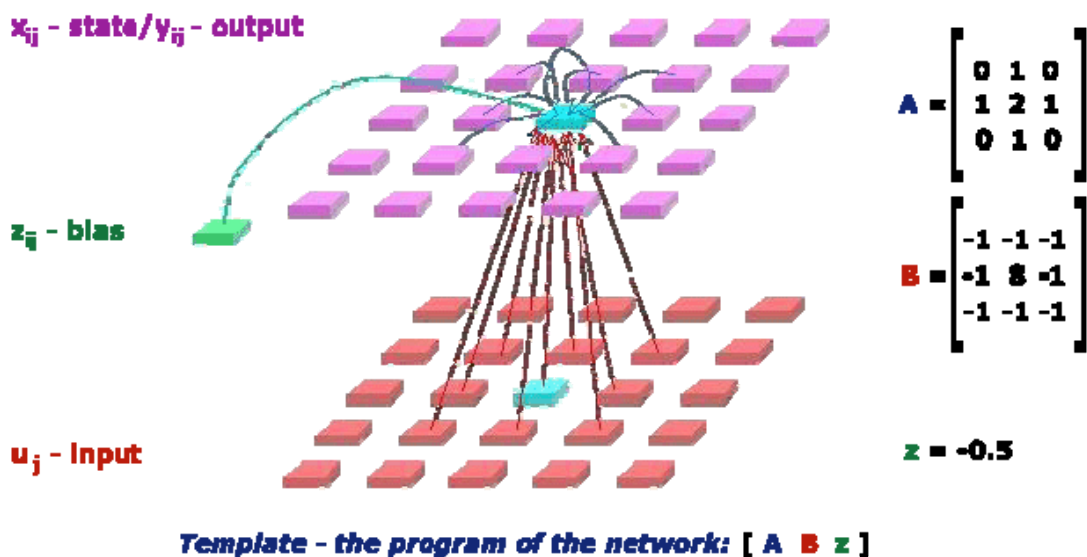
Chip tế bào thị giác [34]

Mạng neuron tế bào được thiết kế giống như mạng lưới gồm có số lượng lớn các tế bào xử lý tương tự giống hệt nhau. Những tế bào xử lý này được sắp xếp trong một cấu trúc đều (thường là hình chữ nhật hoặc lục giác) và được kết nối cục bộ (liên kết với tế bào bên cạnh gần nhất) để tạo ra mối liên kết trên mạch bán dẫn. Mô hình trọng số của các liên kết là một không gian bất biến, nghĩa là mạng chỉ có một số thông số tự do (khoảng vài chục thông số). Mỗi tế bào xử lý được gắn với một cảm biến ánh sáng và có thể xử lý các điểm ảnh có giá trị mức xám (grayscale) hoặc nhị phân. Trên cơ sở cấu trúc bên trong và chức năng của các phần tử xử lý, mô hình liên kết các tế bào và trọng số của các liên kết, một số lượng lớn các kiểu mạng tế bào xử lý ảnh khác nhau có thể thực hiện được.



Hình 20: Mạng CNN 2D

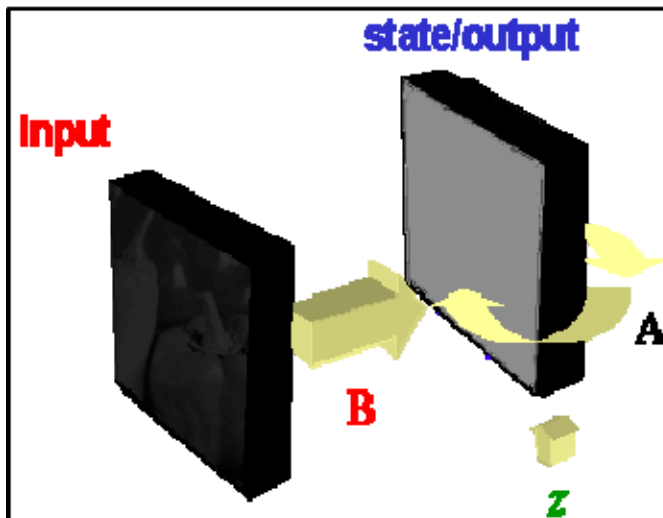
Sau đây là một kiểu kiến trúc mạng neuron tế bào xử lý ảnh thông dụng nhất (Visual Cellular Neural Network). Trong kiến trúc này, phần tử xử lý chứa 2 lớp gồm một lớp vào và một lớp ra:



Hình 21: Mô hình kết nối của mạng neuron tế bào.

Lớp đầu vào cung cấp mỗi liên kết feed-forward cho lớp đầu ra. Trong lớp đầu ra, có các trọng số phản hồi ngược liên tục trong thời gian từ các tế bào lân cận trong không gian.

Một mạng nơ-ron tế bào xử lý ảnh CNN có thể được khởi tạo với 2 ảnh, một ảnh được



nạp vào đầu vào input, ảnh còn lại là đầu ra output. Trong pha xử lý này, ảnh trong đầu ra được thay đổi và ảnh cuối cùng trên lớp ra là kết quả của sự tính toán.

Trong chip tế bào thị giác, ngoài cấu trúc mạng nơ-ron tế bào, còn có một số các kiến trúc thực hiện tính toán hiệu năng cao khác tồn tại. Các kiến trúc này có thể chỉ xác định mạng 1 lớp với liên kết tuyến tính hoặc các kiểu liên kết phi tuyến khác. Ngoài ra còn có các chức năng xử lý toàn cục khác.

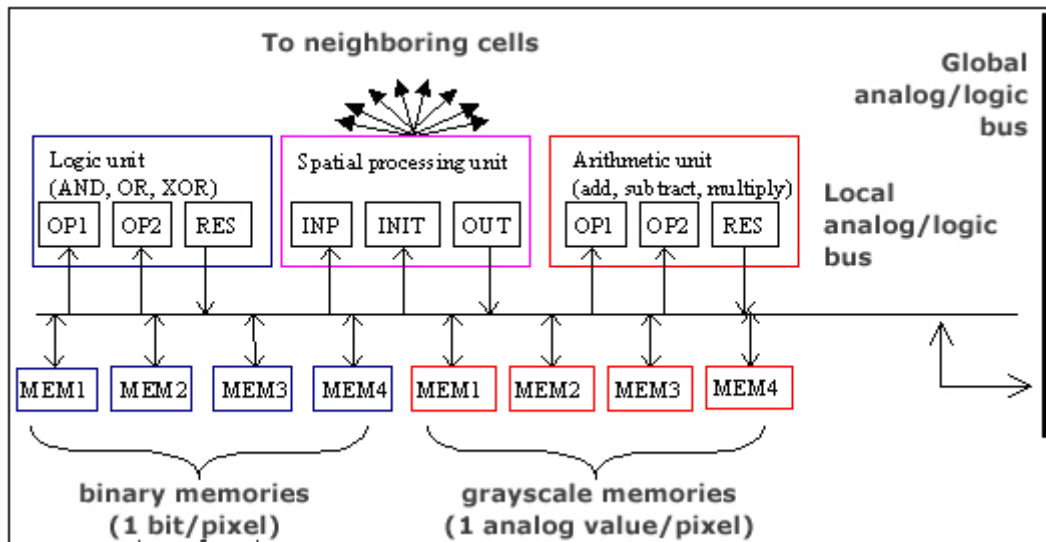
Hình 22: Quá trình tính toán của mạng

nơ-ron tế bào

Giáo sư Tamás Roska từ Viện MTA SzTAKI Budapest, Hungary và giáo sư Leon O. Chua từ Đại học Berkeley California USA đã đưa ra giải pháp để chuyển đổi cấu trúc mảng gồm các tế bào xử lý sang kiến trúc chip xử lý có khả năng lập trình theo thuật toán. Hai giáo sư đã đề xuất giải pháp tạo ra sự liên kết các trọng số khả trình và thực hiện trong các vùng nhớ ảnh xám và ảnh nhị phân. Họ cũng đề xuất thêm thuật toán khả trình và các đơn vị xử lý logic ở mỗi tế bào. Việc lập trình các trọng số và tích hợp việc xử lý và lưu giữ các khối đã tạo ra chip xử lý ảnh trên nền mạng CNN. Một ảnh với kích thước mảng (array-sized image) (mỗi tế bào xử lý tương ứng với một điểm ảnh – image pixel) có thể nạp (download) và lưu trữ trong chip và thực hiện tuần tự những phép tính trên chip. Những kết quả bên trong có thể được lưu trữ và dùng trong thuật toán đang thực hiện. Ví dụ một vài thuật toán xử lý ảnh thay đổi những ảnh có giá trị mức xám gốc thành ảnh nhị phân. Sau khi chuyển đổi, ảnh có thể được lưu trữ trong bộ nhớ nhị phân, phân tán trong chip (on-chip) và các phép toán logic cũng có thể được chấp nhận.

Nếu chúng ta gắn thêm cảm biến quang học (optical sensor) vào mỗi phần tử xử lý, chúng ta nhận được chip xử lý thị giác. Với khả năng tích hợp cảm biến quang học, chip xử lý thị giác có thể bắt ảnh và xử lý chúng ngay lập tức, song song trong từng điểm ảnh. Như vậy ta đã tạo ra một thiết bị xử lý ảnh tốc độ xử lý ảnh rất nhanh (lên tới >10000 Fps) mà lại nhỏ, giá thành rẻ và tiêu thụ điện năng thấp.

Kiến trúc bên trong của một tế bào xử lý được mô tả trong hình 23.



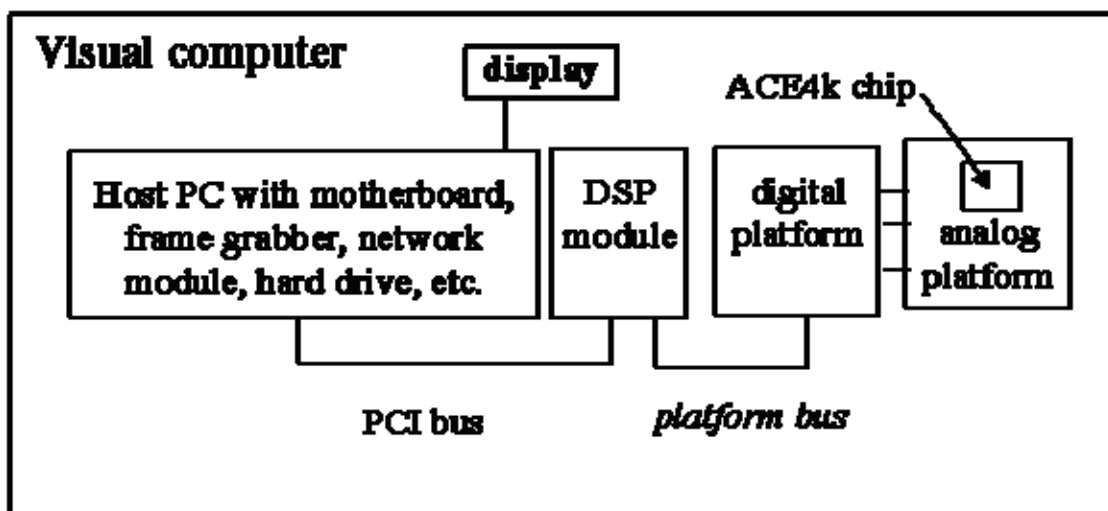
Hình 23: Kiến trúc bên trong của một tế bào xử lý

Máy tính thị giác (Visual computers)

Máy tính thị giác là môi trường tính toán xử lý ảnh chuyên nghiệp hiệu năng cao (high-performance) của các loại chip thị giác. Nó được thiết kế với hai mục đích chính. Một mặt nó là môi trường phát triển của chip thị giác và mặt khác nó là môi trường ứng dụng mà có thể được sử dụng trong công nghiệp.

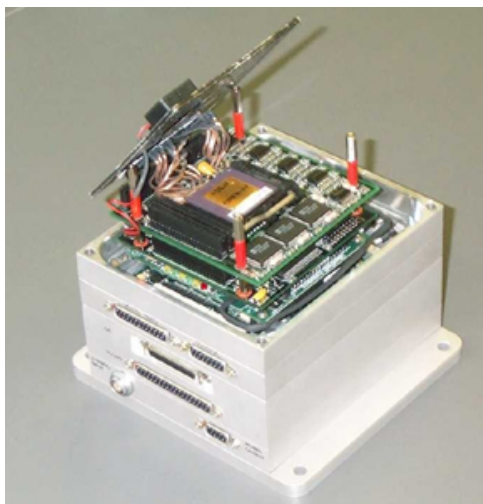
Các máy tính thị giác đầu tiên được thực hiện trên các chip thị giác ACE4K

Kiến trúc của máy tính thị giác (có thể xem ở hình 24) có một chip thị giác ACE4K và một module DSP trong hệ thống. Module DSP này có ba vai trò quan trọng. Đầu tiên, nó cung cấp giao diện PCI nhanh đối với PC công nghiệp chủ hoặc PC để bàn. Thứ hai, nó điều khiển bộ vi xử lý tế bào thị giác ACE4K, cung ứng nó với dữ liệu và điều khiển sự hoạt động của nó. Vai trò thứ ba là DSP mạnh (Texas 320C6202 @ 250 MHz) có thể thực hiện các phép xử lý ảnh mà không thể được thực hiện hiệu quả trên bộ vi xử lý tế bào thị giác CNN.



Hình 24: Sơ đồ khối của máy tính thị giác

Hệ thống được điều khiển bởi PC chạy dưới Windows NT hoặc Windows 2000. hoặc máy tính công nghiệp PC104-plus. Trong trường hợp sử dụng trong phòng thí nghiệm hoặc phòng sạch, hệ thống có thể cài vào máy tính để bàn.



Hình 25: Máy tính thị giác trên PC104-plus Hình 26: Máy tính thị giác trên PC để bàn

Máy tính thị giác có thể xử lý các ảnh video online với các mô đun thu thập ảnh nhúng.

Hệ phần mềm xử lý ảnh (Visual Computer Stack) có thể được lập trình qua ngôn ngữ C và những ngôn ngữ máy (AMC, Alpha) của chip thị giác.

Máy tính thị giác Bi-I [38] [5]

Cấu trúc phần cứng Bi-I.

Hệ thống xử lý ảnh nhanh Bi-I là thế hệ máy tính thị giác CNN kế tiếp. Bi-I ứng dụng công nghệ mạng nơ-ron tế bào bao gồm một sensor nhận ảnh CMOS 1.3 Mpixel và chip xử lý ảnh CNN tốc độ cao ACE16K. Phần tử xử lý nối tiếp của Bi-I là bộ xử lý tín hiệu số tốc độ cao DSP. Nó cũng có một bộ xử lý truyền thông hỗ trợ các giao diện khác nhau. Trong đó, giao diện quan trọng nhất của Bi-I là Ethernet 100 Mbit. Chương trình chạy trên Bi-I được nạp qua Ethernet và máy tính chủ có thể đọc, ghi từ Bi-I qua Ethernet. Hệ thống thị giác Bi-I có thể bắt và xử lý ảnh tới 50000 ảnh với độ phân giải 128x128 pixels trong một giây. Bi-I được trang bị một hoặc hai cảm biến. Hiện nay, có hai cảm biến ảnh có thể lắp vào hệ thống Bi-I : cảm biến ảnh CMOS 1.3Mpixel và chip xử lý ảnh CNN ACE16k_V2 rất nhanh (sản phẩm của AnaFocus Ltd.) . Chip ACE16K_V2 này có khả năng bắt và xử lý ảnh 128x128 pixel với tốc độ cực cao hoặc có nhiệm vụ như bộ xử lý, phân tích các thành phần của ảnh bắt được bằng các cảm biến khác. Bi-I còn có từ 2 hoặc 3 bộ xử lý DSP C6415 Texas Instruments hiệu năng cao chạy với tần số 600MHz có nhiệm vụ như một CPU của hệ thống quản lý các cảm biến trong khi thực hiện việc điều khiển xử lý ảnh rất mạnh. Tùy chọn DSP floating point 150 MHz có thể tăng nhanh sự tính toán đáng kể. Ba bộ xử lý được dành cho thiết bị truyền thông (chạy dưới Linux) chịu trách nhiệm đối với truyền thông nối tiếp, USB và Ethernet.



Đặc điểm kỹ thuật của hệ xử lý ảnh BI-I

- Hoạt động độc lập.
- Tùy chọn chip xử lý ảnh CNN có khả năng bắt và xử lý ảnh tối đa 50000 frame/s.
- Cảm biến CMOS có độ phân giải cao với tốc độ 40 MHz pixel (mức xám hoặc màu).
- DSP fixed point C6415 Texas Instrument nhúng chạy với tốc độ 600 MHz.
- DSP fixed point C6711 Texas Instrument nhúng chạy với tốc độ 150 MHz.
- Bộ xử lý truyền thông Etrax nhúng.
- Cổng USB và Ethernet 100 Mbit.
- Thư viện phần mềm InstantVision với đầy đủ tiện ích.
- C++ SDK với Texas Code Composer Studio, Windows và Linux API để truyền thông với máy chủ trong khi phát triển ứng dụng.

Các loại máy tính thị giác Bi-I được mô tả trong bảng sau:

Bi-i version	Sensor(s) used
Bi-i v2 HRES	1 CMOS
Bi-i v2 ACE16k	1 ACE16k_v2
Bi-i v2 Combi	1 ACE16k_v2 + 1 CMOS
Bi-i v2 Stereo	2 CMOS

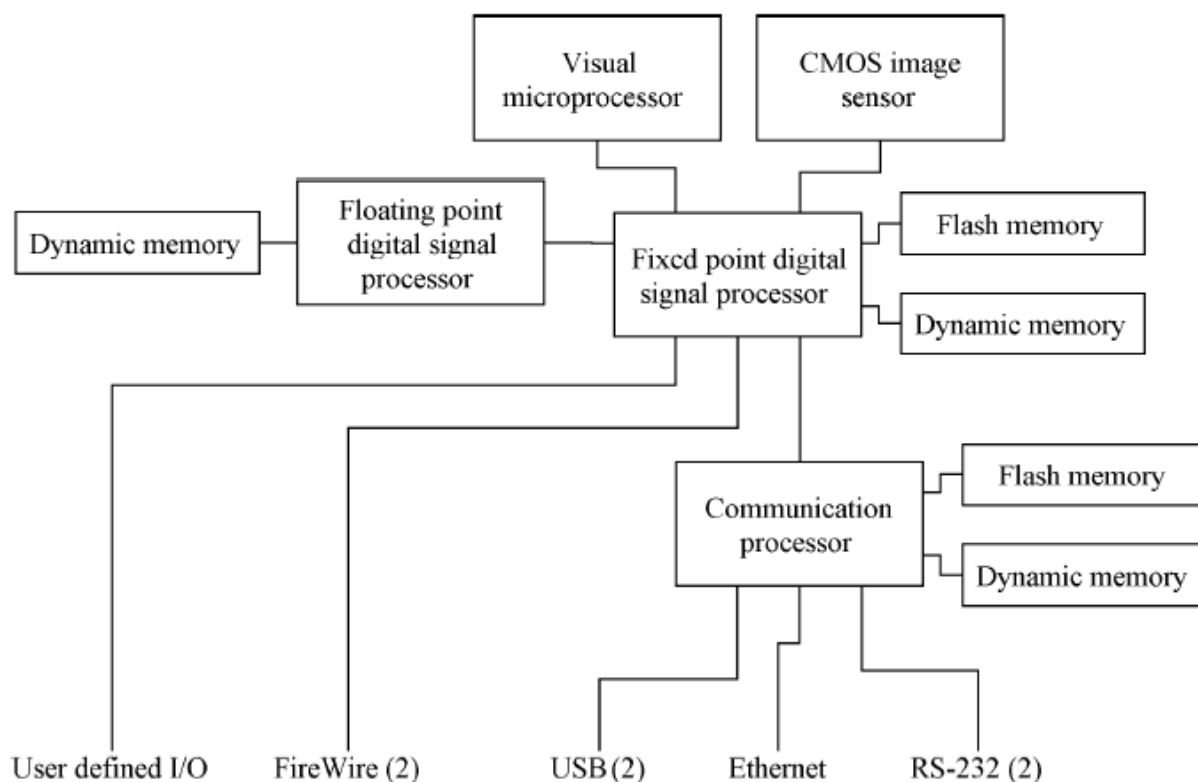
Sensors	
<i>Sensor type</i>	IBIS5A-1300 (Manufactured by Fillfactory), grayscale or color (Bayer pattern)
Resolution	1280x1024 pixel at full resolution, arbitrary ROI windowing, 1:2 electrical zooming capability
Optical format	2/3"
Frame rate	27.5 fps at full resolution; 100 fps at 640x480; 1,000 fps at 128x128; frame rate proportional to image size
Data format	8 bit
Sensor pixel size	6.7x6.7 μm
Dynamic range	64 dB
Minimal integration time	8 μsec
<i>Sensor type</i>	ACE16k_v2 CVM, focal plane array processor
Resolution	128x128 pixel
Optical format	1"
Frame rate (capture and process)	Up to 10,000 frame/sec
Data format	8 bit
Cell size	75.7 x 73.3 μm

Digital processors	
Fixed-point 32-bit CPU	Texas Instruments TMS320C6415 DSP @ 600 MHz with 4800 MIPS processing power
Floating-point 32-bit processor	Texas Instruments TMS320C6711 DSP @ 150 MHz with 900 MFLOPS processing power
Memory	4 MB Flash, 16 MB SDRAM
Communication processor	Etrax 100LX System-on-Chip from Axis

Software	
Basic package	Bi-i software package with development and runtime environment for Windows, Free image processing library
Optional add-ons	API interface for Windows and Linux, SDK, InstantVision software Libraries
Programmability	Native language (included in basic package), C++ using Texas Code Composer Studio

Other features	
Interfaces	100 Mbit Ethernet; 2 USB 1.1; 2 RS232 Serial lines
Optics	C-mount objectives
Data storage	Optional USB flash drive (up to 1 GB)
Power supply	10-28V DC
Power consumption	Bi-i v2 HRES: 6W; Bi-i v2 Stereo: 7W; Bi-i v2 Combi and ACE16k:15W
Dimensions (WxHxD)	154mm x 82mm x 138mm (without optics mounted)
Certifications	CE

Cấu trúc phần cứng của Bi-I



Hình 27: cấu trúc phần cứng của Bi-I

Bi-I có bộ xử lý tín hiệu số hiệu năng cao (DSP) thực hiện các thuật toán tính toán nhanh nối tiếp. DSP có một bộ nhớ flash có thể được sử dụng khi khởi động và một bộ nhớ động được sử dụng trong lúc thực hiện chương trình. Bi-I V1 có một chip DSP TMS320C6202 của Texas Instruments chạy với tốc độ 250 MHz trong khi Bi-I V2 có chip TMS320C6415 chạy với tốc độ 600 MHz.

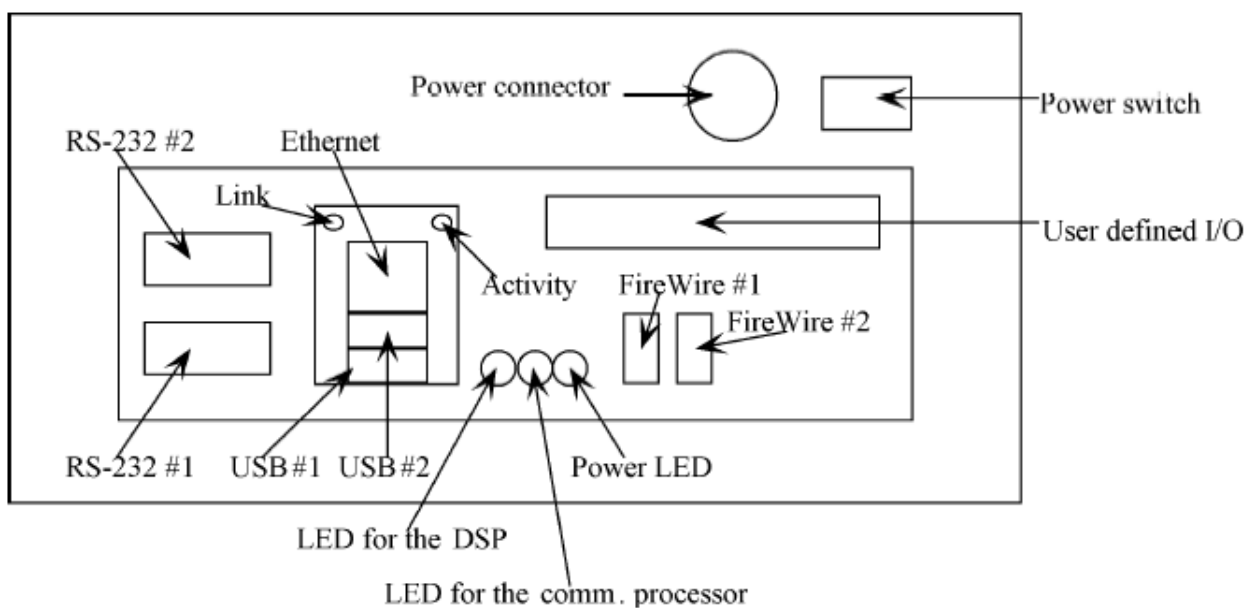
Bi-I có một cảm biến IBIS5 CMOS với 1.3M điểm ảnh IBIS5 của hãng Cypress. IBIS5 có xung đồng hồ điểm ảnh 25 ns, cho thời gian đọc 36ms đối với khung ảnh đầy đủ.

Cảm biến ảnh thứ 2 của BI-I là chip ACE16K với 16k điểm ảnh của hãng AnaFocus. Ltd. ACE16K là bộ xử lý mảng tương tự số CNN. ACE16K được thiết kế đối với các ứng dụng xử lý ảnh nhanh, tốc độ khung có thể đến tới vài chục nghìn ảnh / giây.

Trên thực tế, ACE16K có khả năng xử lý ảnh với tốc độ rất cao với các ảnh có độ phân giải thấp. Ngược lại, IBIS5 có thể xử lý các ảnh có độ phân giải cao với tốc độ trung bình. Cả hai IBIS5 và ACE16K là các thành phần có thể lựa chọn cho những ứng dụng cụ thể. Ngoài ra, những dữ liệu ảnh bên ngoài có thể được kết nối với Bi-I qua kênh FireWire 400 Mbit/s ở hệ Bi-I V2. Giao diện FireWire cũng có thể được sử dụng như ảnh đầu ra. Ví dụ, những ảnh thu được từ cảm biến ACE16K hoặc IBIS5 có thể được truyền đến PC.

Bi-I có một bộ xử lý truyền thông riêng biệt. Bộ xử lý truyền thông là phiên bản MCM của ETRAX 100 LX của hãng Axis. MCM (multi chip module) nghĩa là nhân, bộ nhớ flash và bộ nhớ động được xây dựng trong gói giống nhau cho phép thiết kế nhiều khối. Bi-I V1 có ETRAX100 LX MCM 2+8 có 2 MB bộ nhớ flash và 8 MB bộ nhớ động trong khi Bi-I V2 có thể có bộ xử lý truyền thông giống như thế hoặc ETRAX 100 LX MCM 4+16 với 4 MB bộ nhớ flash và 16 MB bộ nhớ động. Bộ xử lý truyền thông hỗ trợ cho các ngoại vi :

- Giao diện Ethernet: Giao diện này được dùng để kết nối Bi-I với máy tính chủ qua TCP/IP. Qua Ethernet, chương trình có thể được nạp vào bộ nhớ của DSP, việc thực hiện có thể bắt đầu và trong lúc thực hiện dữ liệu có thể được ghi vào hoặc đọc ra DSP. Giao diện Ethernet có thể chạy với tốc độ 100 Mbit/s.
- 2 cổng RS-232
- 2 cổng USB: thiết bị nhớ ngoài có thể được kết nối với cổng USB tạo thêm khả năng chứa đáng kể cho Bi-I. ETRAX 100 LX hỗ trợ chuẩn USB 1.1



Hình 28: Giao diện kết nối của Bi-I V2

1.4.2 Hệ phần mềm phát triển Bi-I [07][09]

Hệ phần mềm phát triển cho hệ thống thị giác Bi-I bao gồm 3 thư viện hỗ trợ phát triển ứng dụng:

- AMC (Analogic Macro Code) là ngôn ngữ lập trình macro dành cho Bi-I. Chương trình viết bằng AMC được dịch sang mã nhị phân và được thực hiện trên Bi-I. Phần mềm soạn thảo AMC và trình dịch AMC được cài đặt như là một thành phần của gói phần mềm Bi-I. AMC không được thiết kế cho các project lớn, tuy nhiên dùng AMC rất hiệu quả cho những ứng dụng nhỏ.
- SDK (Software Development Kit) là thư viện lập trình C++ cho việc phát triển ứng dụng Bi-I. SDK có thể được dùng trong môi trường phát triển cho DSP được gọi là Code Composer Studio. Bi-I SDK làm việc với thư viện xử lý ảnh của Analogic được gọi là InstantVision. Các chức năng trong SDK được thực hiện trên những thành phần khác nhau của phần cứng Bi-I như thực hiện cảm biến CMOS hoặc gửi dữ liệu về máy tính chủ. Ngược lại, các chức năng trong InstantVision là độc lập trong môi trường phần cứng Bi-I, chúng có thể chạy trong các hệ thống khác khá tốt. Thành phần BaseData của InstantVision rất cần thiết đối với hệ phát triển SDK bởi vì các lớp lưu giữ ảnh được xác định trong BaseData được tham chiếu trong toàn bộ SDK.
- API (Application Program Interface) là giao diện phần mềm cho các giao diện ứng dụng với Bi-I. Bi-I_Run thay thế các ứng dụng tùy biến có thể được phát triển với API. Một ứng dụng dùng API chạy trên máy tính chủ, không chạy trên Bi-I. Để phát triển chương trình ứng dụng ta còn cần một môi trường phát triển trên máy tính để dùng API như Microsoft Visual C++.

<i>Chế độ lập trình</i>	<i>Ý nghĩa</i>	<i>Môi trường phát triển</i>
<i>AMC</i>	<i>Phát triển các ứng dụng đơn giản chạy trên Bi-I</i>	
<i>SDK</i>	<i>Phát triển các ứng dụng chạy trên Bi-I</i>	<i>Code Composer Studio</i>
<i>API</i>	<i>Phát triển các ứng dụng chạy trên máy tính chủ</i>	<i>Tích hợp thư viện API vào môi trường phát triển ứng dụng trên máy tính chủ như Microsoft Visual C++</i>

- Việc phát triển ứng dụng cho Bi-I được mô tả chi tiết trong cuốn Bi-I Programming Manual và Bi-I Programming Reference.
- Khi làm việc với Bi-I SDK, nên được xem xét các tài liệu InstantVision Manual và InstantVision Reference
- Gói phần mềm Bi-I có 2 phiên bản : standard và development. Gói standard chứa các thành phần phần mềm để viết, dịch và chạy các chương trình AMC. Ngoài ra, gói development cũng hỗ trợ việc lập trình API và SDK. Các thành phần BaseData và ImgLib của InstantVision được cài đặt với phần mềm phát triển Bi-I. Các gói phần mềm Bi-I đòi hỏi Microsoft Windows 2000 hoặc XP.

Giao diện lập trình ứng dụng Bi-I API

Bi-I API là giao diện lập trình được sử dụng trên máy tính chủ để viết ứng dụng làm việc với Bi-I. Ví dụ, ứng dụng chạy trên máy tính chủ truy cập Bi-I API được xem như ứng dụng chủ. Ứng dụng chủ có thể chạy các chương trình trên Bi-I và trao đổi dữ liệu giữa Bi-I và máy tính chủ. Chế độ mặc định, ứng dụng chủ là Bi-I_Run được cài đặt với tất cả các gói phần mềm Bi-I. Nếu ứng dụng đòi hỏi các chức năng vượt quá phạm vi của Bi-I_Run, ta cần dùng thêm Bi-I API. Ví dụ, nếu ta cần tùy biến giao diện người dùng hoặc nếu phải làm những nhiệm vụ khác ngoài sự làm việc với Bi-I (như truy cập cơ sở dữ liệu hoặc các ngoại vi khác).

Bi-I API được thực hiện như một thư viện liên kết động Windows được gọi là Bi-I-API.dll. Nó có thể được dùng với hầu hết các môi trường phát triển ứng dụng có thể làm việc với các thư viện liên kết động. Các hàm chức năng và các định nghĩa được đặt trong file header C++ đơn giản được gọi là Bi-I-API.h. Thư viện import được gọi là Bi-I-API.lib có thể được sử dụng để tích hợp với ứng dụng của bạn với Bi-I-API dưới Microsoft Visual C++. Bi-I API không điều khiển chương trình thực hiện trên Bi-I một cách trực tiếp. Cách hiểu đơn giản là 2 ứng dụng cùng chạy độc lập: một ứng dụng trên Bi-I và ứng dụng còn lại chạy trên máy tính chủ. Chúng có thể gửi các message cho nhau và chúng có thể trao đổi dữ liệu nhưng chúng không liên kết chặt chẽ với nhau.

Trước khi chạy ứng dụng với Bi-I API, cần đặt cấu hình thiết bị với chương trình Bi-I-Config. Có vài cách nhận message từ ứng dụng chạy trên Bi-I. Ta cần phải chọn chế độ làm việc trước khi chạy ứng dụng. Các chế độ message được liệt kê trong bảng TinfoType như sau:

<i>BI-I_INFO_THREAD</i>	<i>Message từ ứng dụng Bi-I sẽ gửi theo luồng (thread)</i>
<i>BI-I-INFO_WINDOW</i>	<i>Message từ ứng dụng Bi-I gửi tới cửa sổ (window)</i>
<i>BI-I-INFO_CALLBACK</i>	<i>Một hàm callback sẽ được gọi khi một message từ Bi-I được nhận. Hàm này phải phù hợp kiểu khai báo Bi-I_Callback.</i>
<i>BI-I_INFO_QUEUE</i>	<i>Message từ ứng dụng Bi-I được gửi vào hàng đợi bên trong mà có thể được kiểm tra trong ứng dụng chủ.</i>

Sau khi đặt cấu hình thiết bị và quyết định chế độ thông báo (messaging mode), ta có thể cho chạy ứng dụng trên Bi-I một cách đơn giản là gọi hàm Bi-I-Run với các thông số:

<i>Device</i>	<i>Số thiết bị mà được thiết lập với Bi-IConfig</i>
<i>App</i>	<i>Chương trình DSP được thực hiện trên Bi-I. Truyền tham số NULL hoặc chuỗi rỗng nếu muốn chạy chương trình DSP mặc định mà được thiết lập trong Bi-IConfig. Nếu không cho đường dẫn đầy đủ, hệ thống sẽ tìm kiếm sự liên quan đến folder bin trong phạm vi thư mục cài đặt của phần mềm Bi-I</i>
<i>BinFile</i>	<i>Tên của file ABC. Chỉ liên quan khi dùng AMC, truyền tham số NULL hoặc chuỗi rỗng khi chạy một ứng dụng tạo với Bi-I SDK.</i>
<i>DefDir</i>	<i>Mặc định thư mục dữ liệu. Đối với tất cả các hoạt động chủ trên Bi-I (AMC hoặc SDK), các file được đặt tương ứng với thư mục này, trừ khi chúng được cho với đường dẫn đầy đủ. Truyền tham số NULL hoặc chuỗi rỗng nếu không muốn dùng tham số này.</i>
<i>InfoType</i>	<i>Chỉ dẫn chế độ của việc điều khiển message như danh sách ở trên.</i>
<i>InfoParam</i>	<i>Tham số của việc điều khiển message tùy theo từng trường hợp, xem ở bảng dưới.</i>

<i>InfoType</i>	<i>Kiểu InfoParam</i>	<i>Ý nghĩa</i>
<i>BI-I_INFO_THREAD</i>	<i>ThreadId</i>	<i>Xác định luồng mà các message được qua.</i>
<i>BI-I_INFO_WINDOW</i>	<i>WindowHandle</i>	<i>Handle của cửa sổ mà các message được truyền qua.</i>
<i>BI-I_INFO_CALLBACK</i>	<i>Callback</i>	<i>Pointer trỏ đến hàm callback</i>
<i>BI-I_INFO_QUEUE</i>	<i>Bỏ qua</i>	

InfoParam được truyền tham số phụ thuộc vào InfoType

Giá trị trả về của Bi-I_Run là handle có thể được dùng cho vài hàm khác. Các bước chạy chương trình như sau:

- Gọi hàm Bi-I_Init có các tham số chính xác giống như Bi-I_Run. Chỉ khác nhau là nó chỉ nạp ứng dụng vào Bi-I nhưng nó không khởi động ứng dụng.
- Gọi hàm Bi-I_Start để khởi động ứng dụng
- Gọi hàm Bi-I_Close để đóng kết nối với Bi-I

Các message giữa ứng dụng chủ và Bi-I có các thành phần :

<i>Identifier</i>	<i>Giá trị xác định vai trò của message.</i>
<i>Parameter</i>	<i>Chỉ rõ thông tin thêm về message. Nghĩa chính xác phụ thuộc vào giá trị của kiểu Identifier.</i>
<i>Additional parameter</i>	<i>Luôn luôn chứa handle (giá trị trả về của Bi-I_Run hoặc Bi-I_Init) khi nhận các message từ Bi-I).</i>

Nhận các message

Khi dùng chế độ BI-I_INFO_THREAD hoặc BI-I_INFO_WINDOW, các message có thể được nhận với gọi Windows API GetMessage hoặc PeekMessage. Cả hai hàm nhận message trong cấu trúc MSG. Các thành phần identifier, parameter và additional parameter được đặt trong từng thành viên “message”, “wParam” và “lParam”.

Khi dùng chế độ BI-I_INFO_CALLBACK thì tất cả các thành phần message được truyền qua hàm callback.

Khi dùng chế độ BI-I_INFO_QUEUE thì hàng đợi message bên trong có thể được kiểm tra với các hàm Bi-I_PeekMessage và Bi-I_GetMessage. Sự khác nhau giữa chúng là Bi-I_PeekMessage trả về ngay lập tức trong khi Bi-I_GetMessage chờ message. Cả hai chỉ nhận tham số identifier, parameter. Tham số additional parameter không được nhận bởi vì handle mà nhận additional parameter được truyền qua đối số đầu tiên của các hàm này. Khi làm việc với các hệ thống Bi-I trong cùng thời gian, thì hàng đợi message đối với từng hệ thống phải được kiểm tra trong việc gọi hàm riêng biệt.

Ứng dụng chủ có thể nhận message từ Bi-I :

<i>BI-I_MSG_START</i>	<i>Gửi sau khi gọi Bi-I_Run khi ứng dụng khởi động trên Bi-I. Không gửi khi ứng dụng được khởi động với Bi-I_Start hoặc nếu có lỗi trước khi khởi động ứng dụng. Ví dụ, Bi-I không có hiệu lực trên mạng, BI-I_MSG_START không có tham số.</i>
<i>BI-I_MSG_PUT</i>	<i>Bi-I gửi dữ liệu về ứng dụng chủ</i>
<i>BI-I_MSG_CLR</i>	<i>Ứng dụng chủ có thể xóa dữ liệu bên trong của nó</i>
<i>BI-I_MSG_GET</i>	<i>Bi-I yêu cầu dữ liệu từ ứng dụng chủ.</i>
<i>BI-I_MSG_RELEASE</i>	<i>Được dùng trong việc liên kết với BI-I_MSG_GET</i>
<i>BI-I_MSG_WAIT_BEGIN</i>	<i>Chỉ ra rằng ứng dụng Bi-I vào trạng thái chờ mà có định số lượng trễ. Độ dài của trạng thái chờ là ms được truyền qua parameter</i>
<i>BI-I_MSG_WAIT_END</i>	<i>Chỉ ra trạng thái chờ được kết thúc trên Bi-I. Message này không có tham số</i>
<i>BI-I_MSG_STOP</i>	<i>Gửi khi ứng dụng trên Bi-I dừng. BI-I_MSG_STOP không có tham số.</i>

Gửi các message

Dùng hàm Bi-I_PostMessage gửi các message đến Bi-I:

<i>BI-I_MSG_PAUSE</i>	<i>Bi-I tạm dừng việc thực hiện chương trình</i>
<i>BI-I_MSG_RESUME</i>	<i>Bi-I khôi phục việc thực hiện chương trình (sau khi tạm dừng với tham số BI-I_MSG_PAUSE)</i>
<i>BI-I_MSG_WAIT_END</i>	<i>Dừng khi BI-I_MSG_WAIT_BEGIN đã được nhận. Bắt buộc ứng dụng Bi-I kết thúc trạng thái chờ ngay lập tức không có sự chờ kết thúc của sự chỉ rõ việc trễ.</i>
<i>BI-I_MSG_STOP</i>	<i>Dừng việc thực hiện của chương trình trên Bi-I. Bi-I gửi trở lại message giống như thế khi chương trình sẵn sàng dừng.</i>
<i>BI-I_MSG_ABORT</i>	<i>Bỏ qua chương trình trên Bi-I. Dừng message này với sự cần trọng. Cách bình thường để dừng ứng dụng là dùng message BI-I_MSG_STOP. Không dùng handle nhận từ Bi-I_Run sau khi gửi message này.</i>

Truyền dữ liệu

Truyền dữ liệu giữa Bi-I và ứng dụng chủ trên cơ sở cấu trúc Bi-I_HostData. Mô tả dữ liệu trên Bi-I dùng những kiểu dữ liệu được xác định trong thư viện InstantVision. Do đó, một số phần tử của Bi-I_HostData cũng dùng những kiểu dữ liệu này. Cấu trúc Bi-I_HostData có các thành phần:

<i>Type</i>	<i>Kiểu dữ liệu được lựa chọn từ bảng liệt kê InstantVision::TdataType</i>
<i>Identifier</i>	<i>Một số nguyên nhận biết dữ liệu. Trong Biiun, nó là số lượng của số (1,2,...). Trong ứng dụng, bạn có thể dùng số này để cho một số kết quả.</i>
<i>Anonymous union</i>	<i>Một tập hợp mà thực tế chứa dữ liệu trong các trường khác nhau phụ thuộc vào thành viên Type. Xem bảng sau:</i>

Type	Union member	Representation
DATA_TYPE_BIT	Data	unsigned int
DATA_TYPE_BYTE		
DATA_TYPE_INT		
DATA_TYPE_FLOAT	fData	float
DATA_TYPE_XXX_MATRIX	DataPtr	InstantVi- sion::TMatrixHeader*
DATA_TYPE_RGB		
DATA_TYPE_IMAGE	Dib	void*
DATA_TYPE_TEXT	StrPtr	char*

Các message truyền dữ liệu

Dữ liệu từ Bi-I được nhận trong ứng dụng chủ với message BI-I_MSG_PUT. Thông số của message là một con trỏ (pointer) tới cấu trúc Bi-I_HostData. Khi nhận message này, gọi hàm

Bi-I_SetDataEvent hiển thị yêu cầu của máy chủ đã xử lý. Ta có thể giải phóng bộ nhớ cấp phát cho Bi-I_FreeHostData, nếu không cần nhận dữ liệu nhiều hơn.

Message BI-I_MSG_GET được gửi đến ứng dụng chủ khi nó cần gửi dữ liệu đến Bi-I. Thông số của message này là con trỏ tới cấu trúc Bi-I_HostDataRequest gồm có trường Identifier và Type, một con trỏ tới cấu trúc Bi-I_HostData. Ứng dụng chủ phải thiết lập con trỏ này tới giá trị hợp lệ nếu nó có thể cung cấp dữ liệu được yêu cầu hoặc NULL nếu nó không thể cung cấp dữ liệu được yêu cầu. Dữ liệu cung cấp phải phù hợp với dữ liệu yêu cầu

Truyền dữ liệu trực tiếp

Ứng dụng chủ có thể đọc và ghi trực tiếp vào bộ nhớ của DSP ngay cả khi không có các message từ Bi-I được nhận.

Gọi hàm Bi-I_WriteData để ghi vào bộ nhớ Bi-I. Một địa chỉ vật lý và một cấu trúc Bi-I_HostData được truyền qua hàm này. Khi viết dữ liệu với kích thước động phù hợp kiểu của đối tượng xuất phát từ TancestorMatrix với chính xác kích thước giống thế phải tồn tại trong bộ nhớ Bi-I tại vị trí đã cho trước khi gọi Bi-I_WriteData. DATA_TYPE_IMAGE có thể được viết cho TbitMatrix hoặc TRGBMatrix độc lập độ sâu màu. DATA_TYPE_TEXT được chuyển đổi sang TString mà không cần có thực tế kích thước giống như thế: kích thước lớn có thể lớn hơn.

Gọi hàm Bi-I_ReadData để đọc bộ nhớ Bi-I. Địa chỉ vật lý và cấu trúc Bi-I_HostDataRequest được truyền qua hàm này. Trường Identifier của cấu trúc được bỏ qua trong trường hợp này. Trường HostData được làm đầy với dữ liệu được yêu cầu phù hợp kiểu truyền qua. Khi đọc dữ liệu với kích thước động, sự phù hợp kiểu của đối tượng xuất phát từ TancestorMatrix phải tồn tại trong bộ nhớ Bi-I tại vị trí đã cho trước khi gọi Bi-I_ReadData.

Thư viện lập trình Bi-I SDK

Bi-I SDK là tập hợp các thư viện C++ được dùng để lập trình Bi-I. Một số phần của Bi-I SDK trên cơ sở các lớp được xác định trong module BaseData của các thư viện InstantVision. Xem chi tiết ở InstantVision User's Guide và InstantVision Reference.

Môi trường phần mềm

Bi-I SDK được thiết kế để sử dụng với Code Composer Studio từ Texas Instruments (<http://www.ti.com>). Yêu cầu phiên bản là 2.2 hoặc cao hơn.

Chạy ứng dụng

Để chạy ứng dụng trên Bi-I, thiết bị phù hợp đầu tiên phải được cấu hình với Bi-IConfig, sau đó ứng dụng chủ được quy cho số thiết bị như thiết lập trong Bi-IConfig. Bi-IRun là ứng dụng chủ làm việc với gói phần mềm Bi-I. Tùy biến ứng dụng chủ có thể được tạo với Bi-I API. Cả hai Bi-IConfig và Bi-IRun được chỉ rõ chi tiết trong Bi-I Manual.

Khi cấu hình thiết bị chạy ứng dụng, hãy xem dưới đây:

- Trường 'Net' và 'Authentication' phải hợp lệ để có thể kết nối và đăng nhập vào Bi-I.
- Trường 'DSP program' phải trỏ đến ứng dụng được thực hiện. Tùy biến ứng dụng trên cơ sở Bi-I API có thể là mặc định nhưng Bi-IRun không thể làm nó.
- Giữ các box 'ABC file required', 'Debug with JTAG' và 'Load program from Flash' không kiểm tra (unchecked).

- Giữ trường ‘Log file’ để trống trừ khi bạn sẵn sàng muốn ghi cách chạy Bi-I tại mức Driver.
- Các thiết lập ‘Engine’ không có kết quả trong trường hợp này.

Cách dùng Memory

Bi-I V1 có TMS320C6202 DSP với bộ nhớ chương trình bên trong 256K và bộ nhớ dữ liệu bên trong 128KB. Ngoài ra, 8MB của bộ nhớ ngoài cũng có thể được dùng.

Bi-I V2 được xây dựng với TMD320C6415 DSP có 1 MB của bộ nhớ trong mà có thể được dùng như bộ nhớ dữ liệu hoặc chương trình. Dung lượng của bộ nhớ ngoài là 16MB.

Cả hai hệ thống bộ nhớ ngoài có thể là rỗng được dùng cho chứa dữ liệu hoặc chương trình thực hiện, do đó chương trình có thể chạy chậm hơn. Thông thường, giữ mã tới hạn thời gian và dữ liệu thuộc về bộ nhớ bên trong. Giữ trong bộ nhớ mà sao chép dữ liệu vào bộ nhớ trong, chạy một số phép tính và nạp kết quả trở lại bộ nhớ ngoài có thể nhanh hơn tính toán trực tiếp trên dữ liệu trong bộ nhớ ngoài.

Các vector ngắt

Bảng vector ngắt chiếm giữ 128 32-bit words đầu tiên của bộ nhớ. Nó gồm có 16 8-word cổng vào, mỗi cổng vào thuộc về 1 ngắt. Bảng vector ngắt thường là một file asm chứa đúng 8 lệnh đối với mỗi cổng vào mà đủ để thực hiện một nhánh cho một vài mã lệnh thực hiện ngắt.

Các lựa chọn Build chương trình.

Thông thường, các lựa chọn build của project có thể được thiết lập tự do nhưng có một vài cách như sau:

- Thiết lập ‘Target version’ là C620X (-mv6200) đối với Bi-I V1 và C64XX (mv6400) đối với Bi-I V2.
- Luôn luôn build một ít mã endian (không dùng – me)
- Thiết lập ‘Use function Subsection’ (-mo) được giới thiệu
- Liên kết mã với thư viện hỗ trợ runtime, thư viện Bi-I SDK và thư viện InstantVision (cơ sở và xử lý ảnh).

	Bi-i v1	Bi-i v2
Debug	rts6201.lib	rts6400.lib
	IVBaseImgDSP62d.lib	IVBaseImgDSP64d.lib
	IVBaseImgDSP62d.lib	IVBaseImgDSP64d.lib
Release	rts6201.lib	rts6400.lib
	IVBaseImgDSP62.lib	IVBaseImgDSP64.lib
	IVBaseImgDSP62.lib	IVBaseImgDSP64.lib

Khởi tạo và đóng Bi-I SDK

Trước khi dùng bất kỳ hàm nào từ Bi-I SDK, hàm Init_Library() phải được gọi. Điều này khởi tạo tất cả các cấu trúc dữ liệu bên trong đối với truyền thông chủ.

Cuối cùng Bi-I SDK phải gọi hàm Program_End() mà gửi message tới ứng dụng chủ chỉ ra sự hoạt động chương trình trên Bi-I đã kết thúc.

Macro 'main' trong Bi-I.h có thể làm khởi tạo hoặc kết thúc: nó thay thế hàm main với sự tuần tự của Init_Library, hàm main gốc và Program_End(). Do đó, ta không cần gọi trực tiếp Init_Library() và Program_End() nếu Bi-I.h được chứa trong file mà hàm main được thực hiện. Xác định DIRECT_MAIN_FUNC cấm cách chạy này.

Cài đặt hệ thống thị giác Bi-I

Các bước cần thiết để tạo một Bi-I sẵn sàng chạy một ứng dụng

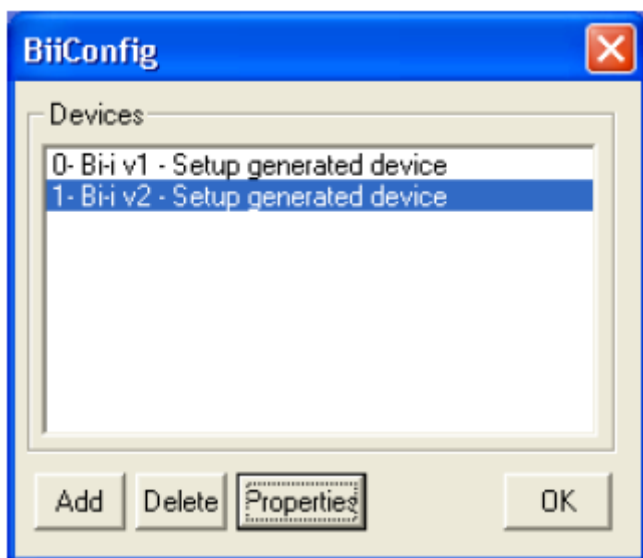
- Kết nối Bi-I với mạng Ethernet và cấu hình nó như mô tả trong phần Network Configuration.
- Cài đặt phần mềm Bi-I trên máy tính. Chạy chương trình cài đặt từ đĩa cài đặt và theo hướng dẫn.
- Cấu hình thiết bị Bi-I trên máy tính như mô tả trong phần Bi-IConfig.
- Kiểm tra hệ thống như mô tả trong phần Testing Bi-I.

Cấu hình mạng cho hệ thống Bi-I

Bi-I được thiết kế chạy trên mạng qua giao thức TCP/IP. Trước khi bắt đầu làm việc với Bi-I, cần đặt cấu hình mạng cho Bi-I trên mạng. Bộ xử lý truyền thông trong Bi-I dùng hệ thống nhúng Linux. Cấu hình mạng của Bi-I được chứa trong file hệ thống của bộ xử lý truyền thông /etc/conf.d/net.eth0.

Bi-IConfig

Dùng công cụ Bi-IConfig để thiết lập cấu hình thiết bị Bi-I. Bi-IConfig được đặt trong Control Panel, nếu hệ điều hành là Windows XP thì chọn Other Control Panel Options. Sẽ xuất hiện cửa sổ :



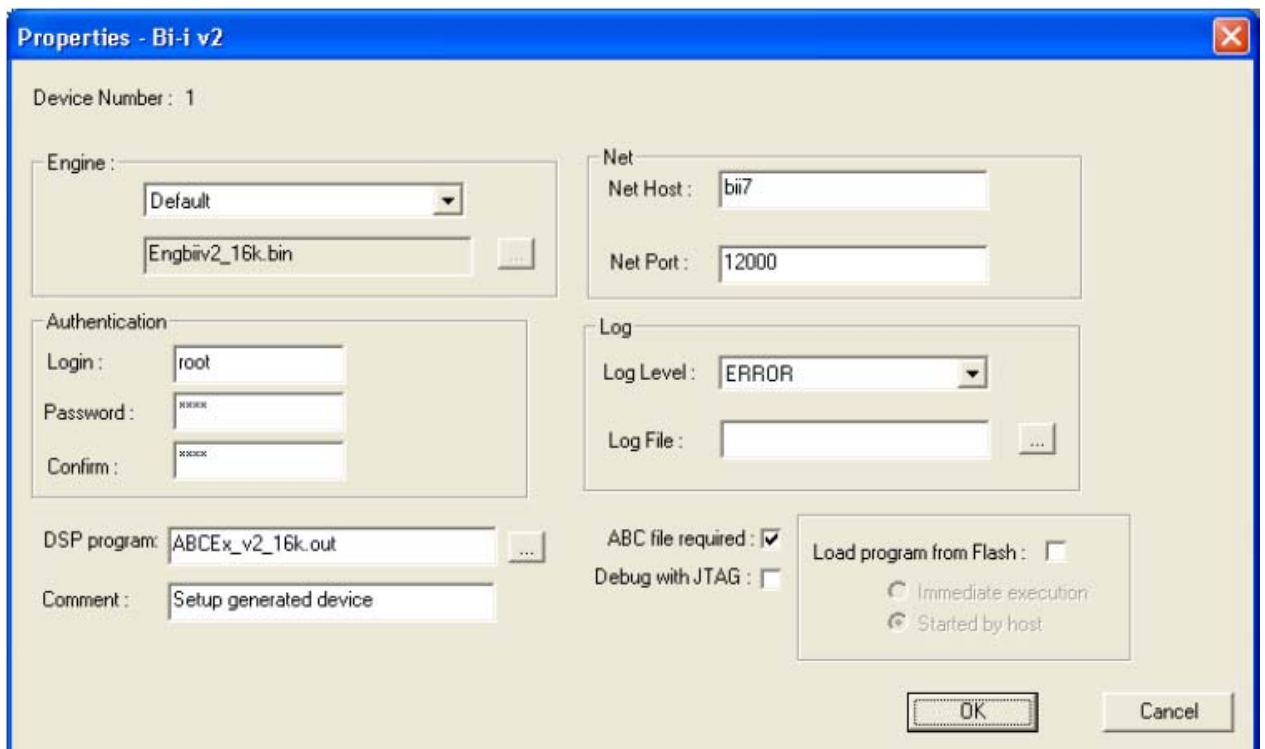
- Add : Thêm thiết bị mới
- Delete : Xóa thiết bị
- Properties: Thiết lập thuộc tính của thiết bị.
- Close : đóng Bi-IConfig.



Thêm thiết bị mới

- Device Number : Số ID xác định Bi-I khi chạy ứng dụng
- Device Type : Phiên bản của Bi-I v1 hoặc v2.

Thuộc tính của thiết bị Bi-I



Engine File engine chứa lệnh AMC. Thiết lập này chỉ áp dụng khi dùng Bi-I với AMC. Thông thường mặc định engine là Default.

Authentication Username và Password khi đăng nhập vào hệ thống Bi-I. Mặc định username là “root”, password là “pass”.

DSP program File chương trình chạy trên DSP trong Bi-I. Khi dùng Bi-I với AMC/ABC file được yêu cầu là ABCEX_v1_16k.out đối với Bi-I v1 và ABCEX_v2_16k.out đối với Bi-I v2.

Net Thiết lập host và port cho Bi-I. Trường host chứa địa chỉ IP của Bi-I hoặc tên tham chiếu tới địa chỉ IP qua DNS hoặc file ‘hosts’. Trường port là 12000 trừ khi cấu hình của bộ xử lý truyền thông đã được thay đổi trên Bi-I.

ABC file required Hiện thị file ABC phải nạp vào bộ nhớ DSP trước khi chạy. Check vào box này nếu Bi-I được chạy với AMC/ABC.

1.4.3. Thư viện xử lý ảnh InstantVision

a. MỞ ĐẦU

Thư viện xử lý ảnh InstantVision là tập hợp các thư viện phần mềm liên kết chéo sau:

- Cấu trúc dữ liệu cơ bản (BaseData)
- Giao diện cho tín hiệu vào/ra (IOComponents)
- Thư viện xử lý tín hiệu và hình ảnh (SI-Lib)
- Thư viện xử lý dòng tín hiệu và chuỗi hình ảnh (SIF-Lib)
- Thư viện phân loại đặc trưng ảnh (FC-Lib)
- Thư viện nhận dạng đa mục tiêu di động (MTT-Lib)

Thư viện InstantVision được sử dụng với phần cứng có TMS320C62xx hoặc TMS320C64xx DSP của Texas Instruments. Thư viện InstantVision cũng phù hợp với nhiều platform khác.

- *Cấu trúc dữ liệu cơ bản* mô tả cấu trúc dữ liệu mà nó cung cấp môi trường C++ cơ bản của InstantVision. Thư viện này bao gồm nhiều loại dữ liệu và cấu trúc khác nhau như dạng bit / byte / RGB / int / float / etc. ma trận, phương pháp biến đổi giữa các cấu trúc và tiện ích logger. Các thành phần khác của gói thư viện đều dựa trên thư viện này.

- *I/O* bao gồm nhóm các lớp truy cập đến các dịch vụ vào/ra. Các lớp này giúp làm đầy cấu trúc BaseData với nội dung từ nhiều loại nguồn dữ liệu và giúp lưu trữ nội dung của các lớp này.

- *I/O Components* mô tả khu vực chủ của truyền thông I/O giữa thiết bị và máy chủ. Cung cấp các đoạn mô tả ngắn về dịch vụ I/O và các lớp dịch vụ cơ bản, từ đó, người sử dụng có thể chia những hệ thống dịch vụ riêng chèn vào các hàm khác nhau khi truyền thông với máy chủ.

- *Thư viện xử lý tín hiệu và hình ảnh (Signal and Image Processing Library)* là tổng thể về thư viện xử lý ảnh (Image Processing Library), nó là tập hợp các lớp và các chương trình con thao tác chung trên ảnh. Thư viện này chứa tác vụ hình thái học và tác vụ mức xám, các chương trình con cho việc tách các đặc trưng và trạng thái tĩnh, các lớp cho việc tách qua video mức xám.

- *Thư viện xử lý dòng tín hiệu và chuỗi hình ảnh* là thư viện các chương trình xử lý ảnh video, tín hiệu và dòng ảnh động

- *Thư viện phân loại đặc trưng ảnh (Feature Classification Library)* là tập hợp các phương pháp phân chia và tách các đặc trưng ảnh khác nhau. Thư viện bao gồm nhiều chương trình con phân loại và tách đặc trưng ở nhiều mức, lựa chọn autonomous và các phương pháp điều chỉnh, nhận biết ảnh, nhận dạng nguyên mẫu và hệ thống ưu tiên dựa trên các thủ tục học off-line, các phương pháp phân chia bao gồm lý thuyết cộng hưởng thích nghi (Adaptive Resonance Theory - ART) và hệ tương tác mờ (Fuzzy Inference System -FIS).

- *Thư viện nhận dạng đa mục tiêu di động* sử dụng khi thực hiện thuật toán thích nghi cho đa mục tiêu di động ngay cả trong môi trường nhiễu. Thư viện này có nhiều phương pháp cho việc tính toán chuyển động như tính khoảng cách, gating, phân chia dữ liệu dựa trên thuật toán Jonker-Volgenant-Castanon (JVC) và đánh giá trạng thái mục tiêu dựa trên bộ lọc phi tuyến Kalman và mô hình đa tương tác(Interacting Multiple Model -IMM).

b. MODULE DỮ LIỆU CƠ BẢN

Module BaseData là một phần của thư viện InstantVision library. Tất cả các chức năng của module BaseData được cài đặt sẵn trong thư mục \InstantVision\BaseData\ bao gồm:

- Các logger có thể gửi thông báo (message) từ một vị trí bất kỳ trong mã chương trình vào thiết bị đích
- Các macro IV_ASSERT, IV_ASSERTOBJ, IV_WARNING và IV_WARNINGOBJ là các công cụ hữu ích cao đối với việc tìm kiếm các lỗi trong khi phát triển ứng dụng và thuật toán.
- Các ma trận

Module BaseData cung cấp các công cụ để lưu trữ dữ liệu theo định dạng uniform. Cơ sở của định dạng này là lớp TAncestorMatrix. Tất cả các kiểu của các ma trận được bắt nguồn từ lớp này. Cấu trúc ma trận gồm có 02 phần. Đầu tiên là cấu trúc TMatrixHeader chứa thông tin về kiểu, kích thước (Dimension) và vị trí (location) của dữ liệu. Thứ hai là vùng nhớ chứa các giá trị của các phần tử của ma trận. Có một vùng không được sử dụng trong bộ nhớ, nếu kích cỡ tính theo byte của một phần tử thuộc ma trận không thể chia hết cho 4, bởi tất cả các hàng đều bắt đầu bằng một word 32 bit.

- Các template biểu thức

Module BaseData cung cấp một tập hợp template C++ (các template biểu thức) cho phép xây dựng các ma trận biểu thức (matrix-expressions). Nó có thể dùng các phép tính thông thường giữa các ma trận và các hằng số. Ví dụ nếu A,B,C,D là ma trận với kích thước như nhau, bạn có thể viết $A=B*C+D$.

- Hỗ trợ các phép tính

Các phép tính được chấp nhận trong các biểu thức bao gồm các phép tính số học, các phép tính Bitwise, Phép toán logic, Các phép tính so sánh và các Các phép tính đặc thù khác

Các công cụ

Module BaseData cung cấp các công cụ cho các tính toán. ma trận như

-Phép biến đổi ma trận

Các kiểu biểu diễn khác nhau của ma trận có thể được chuyển đổi lẫn nhau. Phép biến đổi có thể được thực hiện nếu hai ma trận là tương thích với nhau.

- Các phép Hiệu chỉnh tỷ lệ (scaling)

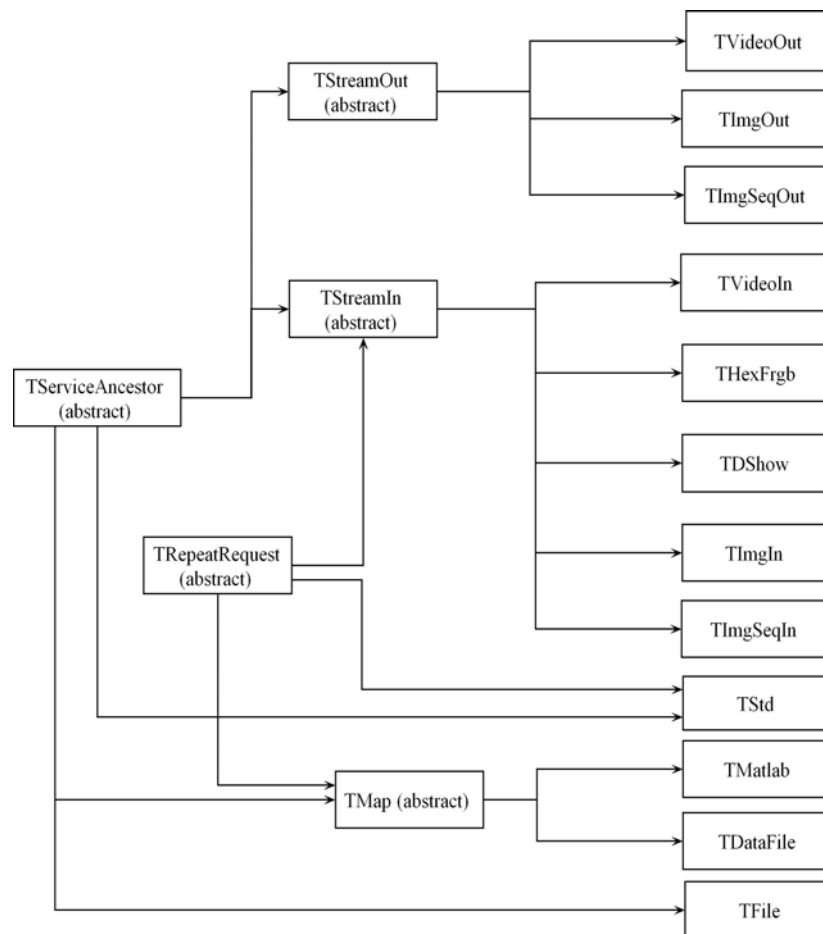
- Tiling

Lớp template Tile là một công cụ phân đoạn một ma trận lớn thành các ma trận con nhỏ hơn không thay đổi về hình thức với phần chồng lên nhau đã xác định và kết hợp các ma trận con này thành một ma trận lớn sau khi xử lý được chứa trong file TTile.h.

c. MODULE VÀO RA

Module vào/ra IO (input/output) là một phần của thư viện InstantVision. Tất cả chức năng của module IO sẵn sàng được dùng trong thư mục include\InstantVision\IO\.

Tất cả các lớp được thừa kế từ lớp cơ sở trừu tượng TServiceAncestor. Lớp này nạp và xóa thư viện điều khiển dịch vụ (service handler library) và điều khiển trạng thái lỗi của lớp (error state). Lớp TStd là duy nhất bởi vì nó không nạp một dịch vụ nào. phần Host chuyển tiếp các hoạt động của lớp TStd tới một mức độ cao hơn, điều này phụ thuộc vào platform mà InstantVision đang chạy



Các chức năng cơ bản của module bao gồm

- Xử lý lỗi
- Input Streams
- Output Streams
- Data File
- File Access
- Matlab

Lớp TMatlab là công cụ hữu ích đối với việc phát triển ứng dụng và thuật toán. Lớp này có thể gửi dữ liệu cho Matlab engine, để có thể được phân tích, kiểm tra, so sánh và mô phỏng dữ liệu, v.v. mà dùng các hàm Matlab. Lớp này mở một Matlab engine. Hàm Set gửi dữ liệu sang Matlab engine. Hàm Get đọc dữ liệu từ Matlab engine. Hàm ExecCommand gửi một chuỗi đến Matlab engine mà sẽ thực hiện lệnh này.

d. MODULE THƯ VIỆN XỬ LÝ ẢNH VÀ TÍN HIỆU (SI-Lib)

Thư viện xử lý ảnh và tín hiệu là một tập hợp các module chương trình C++ tối ưu và các lớp xử lý ảnh và tín hiệu, cùng với các lớp tiện ích để hỗ trợ cho thao tác trên file ảnh và video. Thư viện này đòi hỏi môi trường C++ do lớp BaseData cung cấp. Những module chương trình này thường được dùng trong tính toán chuyên biệt cho ứng dụng thời gian thực, khi mà tốc độ tối ưu để thực hiện lệnh bị hạn chế. Thư viện cũng cung cấp các bổ sung hiệu quả cho hầu hết các kiểu dữ liệu phổ biến trong ứng dụng xử lý ảnh và tín hiệu. và được sắp xếp theo các chức năng khác nhau như sau:

- ✓ “Sử dụng bộ đệm trong” cung cấp một cách nhìn khái quát về cách sử dụng và quản lý bộ đệm trong được sử dụng ở thư viện này
- ✓ “Thao tác Pointwise” cung cấp tổng quan về thao tác pointwise.
- ✓ “Xử lý quang phổ”: các kiến thức cơ bản về chức năng phân tích quang phổ
- ✓ “Lọc tuyến tính”: tổng quan về các thao tác lọc tuyến tính thông thường.
- ✓ “lọc phi tuyến”: nhìn chung về thao tác lọc phi tuyến thông thường.
- ✓ “Hình thái học nhị phân”: khái quát về thao tác hình thái học nhị phân, và đưa ra những mô tả cụ thể về mặt nạ, phép lặp, phép tính thay thế nhị phân, và hình thái học thích nghi.
- ✓ “Hình thái học đa mức xám” nói về các thao tác hình thái học đa mức xám, và mô tả chi tiết về phép tính biên.
- ✓ “Matching mẫu” cho biết về việc matching mẫu và các phép tính thay thế
- ✓ “Phép phân tích Histogram”
- ✓ “Tìm đặc trưng”: cung cấp các chương trình tính đặc trưng
- ✓ “Trích chọn đặc trưng”: là các thường trình trích chọn đặc trưng
- ✓ “Phân tích thống kê” mô tả các chức năng phân tích thống kê đa dạng

Ta hãy điếm qua một số chức năng của thư viện xử lý ảnh và tín hiệu:

Tính toán biên trong các hàm đa mức xám (Grayscale Functions)

SI Lib cung cấp 4 cách tính biên khác nhau:

<i>Không biên</i>	<i>Ảnh đầu vào không có biên chỉ được xử lý đơn giản và việc tính toán biên sẽ bị bỏ qua. Biên sẽ là không xác định</i>
<i>Biên cố định</i>	<i>Ảnh đầu vào không có biên chỉ được xử lý đơn giản, và việc tính toán biên sẽ bị bỏ qua. Trong bước cuối cùng, biên được thiết lập ở một giá trị đã cho.</i>
<i>Tính toán biên (điều kiện đường viền cố định)</i>	<i>Biên sẽ được xử lý với điều kiện có đường viền xung quanh ảnh, tức là các đường viền bao quanh bên ngoài ảnh vào. Điều kiện đường viền này được cho bởi người dùng.</i>
<i>Tính toán biên (copy điều kiện biên)</i>	<i>Biên sẽ được xử lý dựa trên điều kiện biên xung quanh ảnh, tức là các đường viền bao quanh bên ngoài ảnh vào. Điều kiện đường viền ở các góc chính là các góc của ảnh vào</i>

Mặt nạ nhị phân

Nhiều hàm có mặt nạ như một tham số đầu vào. Mặt nạ xác định các pixel nào của ảnh vào sẽ bị bỏ qua hay thêm vào trong quá trình tính toán, thông qua việc tìm kiếm các mẫu pixel nhất định. Các mặt nạ cho các chức năng hình thái học phải chứa các giá trị âm, zero và giá trị dương. Có hai cách diễn giải khác nhau cho các phần tử mặt nạ này:

- ✓ Ở hầu hết các chương trình (như Erosion, Dilation, v.v...), phần tử mặt nạ được chia theo các giá trị dương và không dương: giá trị dương xác định các pixel nào được tính vào trong quá trình thực hiện, còn các phần tử âm và zero nghĩa là bỏ qua pixel
- ✓ Nhưng ở một số chương trình khác như HitAndMiss, Thinning, Thickening v.v..., các phần tử mặt nạ có 3 ý nghĩa khác nhau: các giá trị âm chỉ “false mask”, zero cho biết “DON’T CARE”, còn giá trị mặt nạ dương nghĩa là “true mask”

Các mặt nạ Box và Cross là mặt nạ chung, không nhấn mạnh bất kì hướng nào sau đây:

<i>Mặt nạ Cross</i>	<i>Mặt nạ này chỉ bao gồm 4 vùng lân cận của mỗi pixel trong quá trình tính toán. Mặt nạ Cross:</i> $\begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{matrix}$
<i>Mặt nạ Box</i>	<i>Mặt nạ này bao gồm 8 vùng lân cận của mỗi pixel trong quá trình tính toán. Mặt nạ Box:</i> $\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$

Hầu hết các chức năng đều mang lại khả năng lặp cho đến khi không thể thay đổi được nữa. Điều đó có nghĩa phép lặp sẽ dừng lại khi bức ảnh cuối cùng không thay đổi sau khi lặp. Trong nhiều chức năng hình thái học, giá trị zero của tham số lặp nghĩa là lựa chọn này, trong trường hợp đó, chức năng này sẽ quay lại con số của phép lặp đã thực hiện. Để tránh chu kỳ lặp, có một chức năng watchdog được cài đặt bên trong, cụ thể là nếu số lần lặp đạt đến giá trị trong MORPH_MAX_ITERATION, chức năng này sẽ trở lại trạng thái thực sự của đầu ra và sẽ có một cảnh báo được đưa ra.

Các danh mục chức năng trong thư viện SI-Lib

- Tác vụ pointwise

Nhóm chức năng này bao gồm một tập hợp lớn các hoạt động pointwise như các phép biến đổi ở các mức cường độ phi tuyến khác nhau, bao gồm chuẩn hóa và tạo ngưỡng.

Điểm cốt lõi của tác vụ này được thực hiện trong một lớp dùng một bảng tham chiếu (look-up table) để biến đổi các giá trị cường độ của các pixel vào. Trong lớp này, TByteLut có thể được khởi tạo với các chức năng khác nhau mà các chức năng thực hiện khác nhau trong khoảng [0,255] vào bản thân nó. Thư viện SI cung cấp một số chức năng đối với hoạt động chung nhất được dùng trong xử lý ảnh, như co giãn tương phản (contrast stretching), phân ngưỡng khác nhau (different thresholdings), xóa hoặc trích bit, ...v.v. Tuy nhiên, người dùng có thể thực hiện chức năng của bản thân họ hoặc gán bảng tham chiếu (look-up table) của nó như TbyteMatrix.

- Xử lý quang phổ

Chức năng này bao gồm phép biến đổi Fourier nhanh (FFT) đa thứ nguyên và phép nghịch đảo FFT cùng với nhiều phép xử lý quang phổ khác bao gồm phân tích năng lượng và giai đoạn quang phổ

Để thực hiện nhanh thuật toán FFT, nhiều phép tính có thể được thực hiện một lúc và kết quả sẽ được lưu giữ lại trong pha khởi đầu. Thư viện SI che giấu tất cả các phần bất tiện này bằng cách thực hiện các chức năng của FFT theo tầng lớp khác nhau. Để khởi tạo, chỉ đơn giản là tạo ra một đối tượng Tfft (hay Tfft2 ở dạng 2D) bằng cách qui định kích cỡ của số FFT sẽ được dùng, và rồi đối tượng được tạo ra có thể được dùng như một chức năng để thực hiện quá trình biến đổi.

- Lọc tuyến tính

Nhóm chức năng này kết hợp chặt chẽ giữa hoạt động lọc quang phổ tổng quát bao gồm lọc FIR.

- Lọc phi tuyến

Nhóm chức năng này liên kết chặt chẽ các hoạt động lọc phi tuyến bao gồm giá trị trung bình nhỏ nhất, lớn nhất, trung bình, alpha trim vv...

- Hình thái học nhị phân

Tất cả các chương trình hình thái học đều là các chức năng độc lập. Điều đó có nghĩa là các chức năng không thuộc bất kỳ một lớp nào, không có trạng thái bên trong nào và không thay đổi môi trường trừ ảnh đưa ra. Mỗi phần là một chức năng nhập vào-xuất ra đơn giản, thực hiện hoạt động hình thái học như co giãn ăn mòn, chuyển dịch v.v... trên ảnh nhập vào, và áp dụng kết quả vào ảnh xuất ra.

- Hình thái học thích nghi

Hình thái học thích nghi là một dạng đặc biệt của phép co giãn được điều khiển bởi ảnh xám. Hình thái học thích nghi có thể sử dụng tốt trong việc phân đoạn hình ảnh và xử lý dò tìm khung biên cho nhiều loại hình ảnh cỡ lớn.

- Hình thái học đa mức xám

Hoạt động hình thái học đa mức xám có thể được viết theo thứ bậc từ hai phần tử thực hiện cơ bản, gọi là co giãn và sự ăn mòn. Sự co giãn và ăn mòn được biểu hiện bằng tham số thông qua việc các phần tử được cấu trúc, gọi là các mặt nạ. Những phần tử được cấu trúc này có thể là phẳng (giá trị nhị phân) hay không phẳng (giá trị nguyên). Các chức năng dùng các phần tử cấu trúc phẳng ở trường hợp mặc định. Chúng thay đổi tất cả các số zero hoặc số âm thành giá trị sai, và các số dương thành giá trị đúng trong mặt nạ.

Phân tích biểu đồ Histogram

Nhóm này bao gồm các phép tính biểu đồ Histogram hình ảnh, sự cân bằng và phân tích thống kê.

Dò tìm đặc trưng

Nhóm chức năng này bao gồm các phép tính dò tìm đặc trưng dựa vào các đặc trưng khác nhau của đối tượng như kích cỡ, các biên, kết cấu,...

Trích chọn đặc trưng

Tách đặc trưng là một sự chuyển đổi của thông tin thị giác của một ảnh nhị phân sang định dạng số. Nó được dựa trên việc lập bảng chỉ dẫn cho các tập hợp được kết nối của những pixel tích cực ("đối tượng") trên bức tranh, và sau đó tính toán một vài đặc trưng (như là diện tích, chiều dài, hướng, ...), hoặc để các số liệu nhiều hơn, thông kê chung cho những đặc trưng này trên tất cả các đối tượng.

Thư viện SI cung cấp 03 chức năng riêng biệt để tách những đặc trưng từ các ảnh nhị phân, cụ thể là *CalcFeatures()*, *CalcFeatureStats()*, *CalcSizeStats()*. Để biết chi tiết về cách dùng các chức năng này, hãy tham khảo *InstantVision Reference*.

Đặc trưng của tất cả các đối tượng trên một ảnh nhị phân có thể được tính toán bằng cách sử dụng hàm *CalcFeatures()*. Các đặc trưng sẵn có thể được sắp xếp thành 2 nhóm chính. Những đặc trưng trong nhóm thứ nhất được tính toán trực tiếp từ đối tượng, trong khi nhóm thứ hai bao gồm một vài thuộc tính của hình elíp có cùng mô men thứ hai với đối tượng.

Các đặc trưng được tính toán trực tiếp:

<i>Diện tích</i>	<i>Là diện tích đối tượng, như là số pixel thuộc về đối tượng</i>
<i>Đường kính tương đương</i>	<i>Là đường kính của vòng tròn có chung diện tích với đối tượng</i>
<i>Hộp bao quanh</i>	<i>Là các tọa độ của góc trên cùng bên trái và góc dưới cùng bên phải của hình chữ nhật nhỏ nhất (với các cạnh song song với mép của ảnh) bao quanh đối tượng.</i>
<i>Phạm vi</i>	<i>Là đơn vị đo xem đối tượng lấp đầy khoảng không nó sử dụng như thế nào. Là thương số của diện tích đối tượng và diện tích của hộp bao quanh</i>
<i>Cực trị</i>	<i>Là những điểm cực của đối tượng trên hộp bao quanh.</i>

Những đặc trưng được đo từ hình elíp có cùng moment thứ hai với đối tượng:

<i>Trọng tâm</i>	<i>Là trung tâm của elíp, trùng với tâm hình học của đối tượng</i>
<i>Chiều dài trục lớn</i>	<i>Là chiều dài trục lớn của elíp</i>
<i>Chiều dài trục nhỏ</i>	<i>Là chiều dài trục nhỏ của elíp</i>
<i>Độ lệch tâm</i>	<i>Là tỉ lệ của khoảng cách tập trung và chiều dài trục chính của elíp. Đường tròn có độ lệch tâm là 0, đường thẳng là 1</i>
<i>Hướng</i>	<i>Là góc của trục lớn tương ứng với chiều dương trục hoành</i>

Những đặc trưng được tính toán có thể được thể hiện trên một vectơ kiểu *TFeaturesMatrix*, được đưa ra như một tham số. Chiều dài của vectơ quyết định số lượng tối đa của đối tượng (số lượng đối tượng được tìm thấy được trở lại trong một biến số khác). Mỗi phần tử là một cấu trúc *TObjFeatures* chứa tất cả các đặc trưng có sẵn của một đối tượng.

Thống kê dữ liệu từ những đặc trưng

Có 2 chức năng trong thư viện SI để rút ra các số liệu thống kê khác nhau từ một *TFeatureMatrix*, đầu ra của việc trích chọn đặc trưng của đối tượng.

CalcFeatureStats() tính toán các thuộc tính thông thường của dữ liệu như số lượng nhỏ nhất, lớn nhất, các giá trị trung bình khác nhau, sự dao động, sự nghiêng. Những số liệu thống kê cần thiết có thể được đưa ra dưới dạng vectơ được xây dựng từ những giá trị trong bảng liệt kê của *TSatSelector* cho các đặc trưng. Kết quả nhận được trong một vectơ kiểu *TSatMatrix*, bao gồm một cấu trúc *TSat* như là một phần tử cần thiết cho mọi đặc trưng. Chiều dài vectơ nên bằng với số lượng các đặc trưng mà các số liệu thống kê cần phải tính toán.

Những bộ lọc số liệu thống kê phải được chọn chế độ sắp xếp tăng dần với những bộ lọc đặc tính tương ứng.

CalcSizeStats() rút ra một biểu đồ Histogram gồm 3 phần tử dữ liệu: trả lại những số có giá trị đặc tính nhỏ, trung bình và lớn trên vectơ, nơi giới hạn giữa 3 mức độ là những thông số đầu vào của từng đặc trưng. Kết quả được biểu diễn dưới dạng vectơ *TSizeStatMatrix* bao gồm 1 cấu trúc *TSizeStat* với chức năng là một phần tử cho mọi đặc trưng cần thiết. Chiều dài vectơ có thể bằng số lượng số liệu thống kê kích cỡ đặc trưng cần được tính toán.

Phân tích thống kê

Nhóm chức năng này bao gồm các chương trình để phân tích thống kê trên các dữ liệu vô hướng, như là các phép tính tổng, lớn nhất, nhỏ nhất, trung bình, rms, h-mean, độ nghiêng, ...; trên các chức năng tỉ trọng khả thi như là các phép tính min, max, median, giá trị mong đợi và tạm ước.

Việc thực hiện này chỉ cho phép có một phép tính quét qua các dữ liệu, thậm chí khi phải tính toán nhiều số liệu thống kê hơn.

e. MODULE THƯ VIỆN XỬ LÝ DÒNG TÍN HIỆU VÀ LUỒNG ẢNH (SIF-Lib)

Các chương trình trong thư viện Xử lý dòng tín hiệu và luồng ảnh (Signal and Image Flow Library) được xây dựng trên các thuật toán tối ưu của thư viện Xử lý ảnh và Tín hiệu (SI Lib) để cung cấp giải pháp cho những vấn đề phức tạp hơn. Thư viện này hỗ trợ môi trường C++ trên nền thư viện cơ sở BaseData. Những thuật toán trong thư viện này cũng đã được tối ưu để tận dụng những đặc điểm nổi bật của bộ xử lý. Thư viện SIF gồm những phần chính sau::

- “Đánh giá luồng quang học” được sử dụng để đánh giá luồng quang học (optical flow) trong luồng ảnh (image flow).
- “Bám biên tích cực” cung cấp các kỹ thuật khác nhau cho bám biên của đối tượng động theo không gian- thời gian.
- “Tìm nền” mô tả một thuật toán thống kê tách đối tượng ra khỏi nền.

Đánh giá luồng quang học

Thuật toán đánh giá luồng quang học của một đối tượng chuyển động trong chuỗi ảnh sử dụng phương pháp đồng nhất mẫu (pattern matching).

Luồng quang học là một khái niệm để xét chuyển động của đối tượng được mô tả bằng hình ảnh. Thông thường chuyển động được thể hiện bằng các vectơ có gốc và ngọn tại các điểm trong chuỗi ảnh số. Luồng quang học sử dụng để hỗ trợ nhận dạng mẫu, thị giác máy, và các ứng dụng xử lý ảnh động khác. Nó liên quan mật thiết đến đánh giá chuyển động và bù chuyển động.

Ở đây khái niệm luồng quang học sẽ được sử dụng hoặc để mô tả trường chuyển động của mỗi điểm ảnh với các vectơ tại mỗi điểm (định nghĩa chuẩn) hoặc cho các tập hợp các điểm, ví dụ sử dụng để đánh giá chuyển động trong các phương pháp nén ảnh khác nhau (định nghĩa mở rộng).

Luồng quang học được tính tại các điểm nút lưới của một lưới được định nghĩa trong toàn bộ ảnh (thể hiện lưới phù hợp của quá trình tính toán). Sự tính toán dịch chuyển tại mỗi điểm nút lưới dựa trên đồng nhất mẫu ảnh xám hoặc ảnh nhị phân bởi các phương pháp tìm kiếm và đánh giá cực đại .

Bám biên tích cực [12]:

Các thuật toán bám biên tích cực thích hợp với biên phức tạp và mô tả cấu trúc của các đối tượng khi phân tích một ảnh video.

Phương pháp bám đường biên tích cực đã trở nên công cụ phổ biến trong nhiệm vụ xử lý nhiều ảnh như phân loại, bám và mô hình hoá. Một đường biên tích cực được định nghĩa như một đường cong co giãn mà có thể được điều khiển bởi các đặc trưng ảnh và các giới hạn hình học để thích ứng với biên của đối tượng quan tâm. Kỹ thuật biên tích cực thường được phân loại hoặc là dựa trên mức năng lượng hoặc trên mô hình mức. Phương pháp thứ nhất, mang ý nghĩa vật lý và thể hiện các biên ẩn như các đường cong được thông số hoá trong dạng hàm Lagrange. Phương pháp thứ hai dựa trên thuyết tiến hoá đường cong và ứng dụng các đường cong ẩn như tập mức của hàm bậc cao liên quan đến công thức Euler. Sự phân loại này tuân theo cách thể hiện biên, các ứng dụng và các khả năng xuất phát từ quan điểm hoạt động (đặc biệt, khả năng để quản lý thay đổi trong biên hình học). Tuy nhiên, hầu hết trong số chúng yêu cầu công suất tính toán tương đối mạnh mà giới hạn sử dụng trong các ứng dụng yêu cầu thời gian xử lý nhanh. Nhược điểm này được loại trừ bởi phương pháp mới cho phép mô phỏng các phương trình đường cong, điều này thường dẫn đến một sự thoả hiệp giữa tốc độ xử lý và độ mềm dẻo trong xử lý biên.

Khác với các phương pháp nói trên, mạng CNN tế bào được sử dụng để giải quyết vấn đề thời gian tính toán cao của các kỹ thuật biên tích cực cổ điển. Mạng CNN dựa trên sự rời rạc hoá

giữa mức các điểm ảnh của biên và thực hiện trên một cấu trúc xử lý song song trên mỗi một tế bào tại biên, từ đó dẫn tới một quá trình xử lý ảnh nhanh mà không có sự trả giá về hiệu quả của xác định vị trí biên. Hiện nay, có hai phương pháp bám đường biên tích cực dùng mạng nơ ron tế bào. Một là phương pháp tính toán sóng tích cực. Phương pháp này có kỹ thuật lan truyền theo vùng hình học, ở đó tiến hoá biên được định nghĩa bởi các đường biên của sóng kích hoạt. Do đó, như các phương pháp mô hình mức, tiến hoá biên được thể hiện rõ ràng như lan truyền sóng trước. Phương pháp này đã minh hoạ khả năng mềm dẻo cao trong kỹ thuật phát triển biên và giống như các mô hình ẩn cho các lời giải đơn giản đối với các thay đổi của đối tượng hình học khi hai sóng va chạm nhau. Tuy nhiên, trong kỹ thuật này điều kiện ngừng lan truyền là phức tạp dẫn đến tăng đáng kể độ phức tạp tính toán trong các ứng dụng thực tế. Phương pháp thứ 2 là kỹ thuật mức điểm thể hiện kỹ thuật đường biên tích cực tác động theo hình học. Trong phương pháp này, các đường biên được xác định bởi các thông tin cục bộ và các thành phần hiệu chỉnh phụ thuộc vào các đường cong biên.

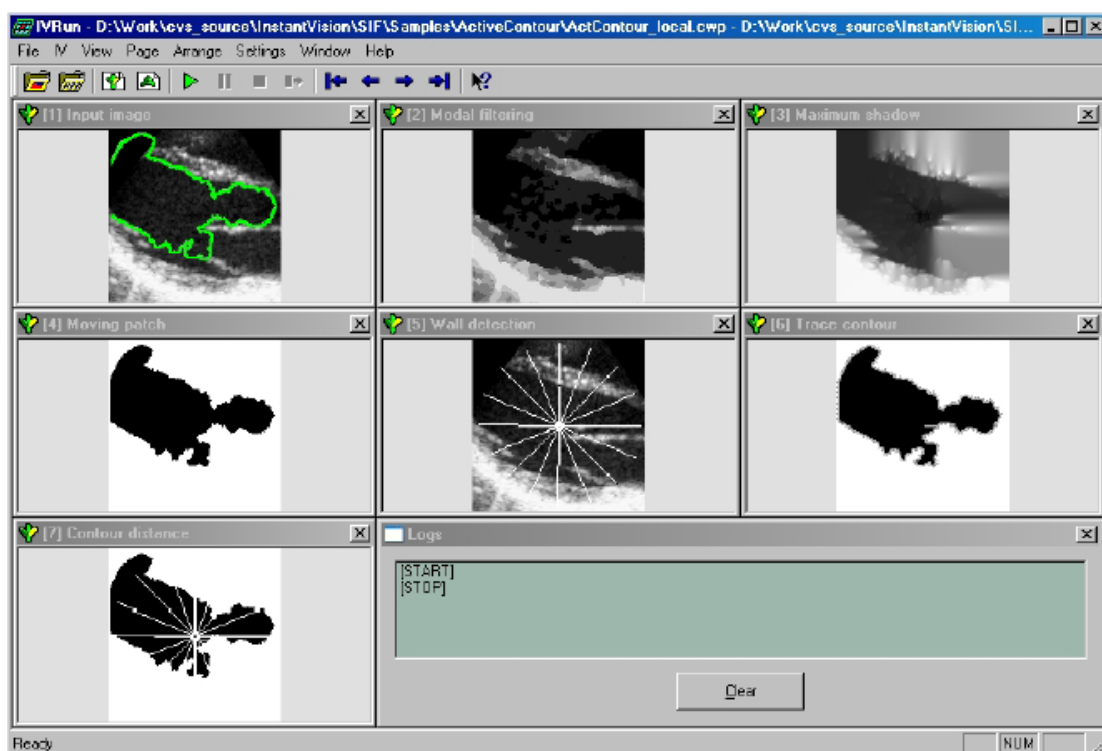
Các hàm chính của kỹ thuật bám biên:

<i>WallDetect</i>	Tìm biên cho một đối tượng có cường độ mức xám thấp so với đường biên có cường độ cao. Việc tìm kiếm tạo ra các đường đồng tâm từ trung tâm của ảnh. Các điểm biên sẽ là giao của các biên thực sự và các đường cong này. Hàm này trả về một kiểu cấu trúc, chứa tọa độ của các đường biên và thông tin thống kê về đường biên.
<i>MaxShadow</i>	Tạo ra một bóng mờ từ một mảng đen đặt giữa trung tâm. Nói cách khác, ứng dụng các thuật toán sử dụng hiệu ứng làm đầy, để tạo ra các bóng mới tại các điểm tương ứng với đường biên của các mảng đen. Nó được sử dụng hiệu quả tạo ra các ảnh đầu vào cho hàm <i>MovingPatch</i> .
<i>MovingPatch</i>	Tìm đường biên của các vùng có cường độ sáng thấp được vẽ bởi các mảng biên có cường độ cao. Đây là hàm tìm biên chính. Nó được sử dụng hiệu quả cho tìm biên ảnh có chất lượng thấp. Để có kết quả tốt nhất, ảnh gốc được xử lý trước bởi vì độ phức tạp của thuật toán phụ thuộc vào số lần xử lý. Một ảnh gốc được chuẩn bị tốt có thể giảm đáng kể số lần xử lý và tăng chất lượng của ảnh đầu ra. Nó cũng quan trọng trong khi chuẩn bị ảnh đầu vào tạo thuận lợi cho tìm đường biên. Ví dụ, sử dụng các hàm <i>MaxShadow</i> hoặc <i>ModalFiltering</i> để tiền xử lý.
<i>ModalFiltering</i>	Hàm này áp dụng các kiểu khác nhau của lọc cục bộ. Nghĩa là tất cả các phương pháp lọc sử dụng một điểm lân cận để tính toán giá trị mới của điểm ảnh. Điều này hiệu quả cho bước tiền xử lý. Ví dụ, một bộ lọc trung bình có thể dễ dàng loại bỏ nhiễu từ tần số cao từ ảnh đầu vào.
<i>MaskPicture</i>	Điền một vùng với màu sắc xác định vào ảnh đầu vào phía trên hoặc phía dưới của một đường cho trước. Nó được sử dụng hiệu quả cho tiền xử lý ảnh đầu vào. Ví dụ, xoá bỏ phần không cần thiết của ảnh đầu vào.
<i>TraceContour</i>	Bám đường biên của mảng đen, và trả về thông tin đường biên như cặp tọa độ. Nó sử dụng hiệu quả cho bước xử lý ảnh. Ví dụ, sử dụng các đường biên đã xác định tọa độ để xác định các kiểu đối tượng được tìm thấy.
<i>ContourDistances</i>	Trả về khoảng cách từ tâm ảnh đến biên của mảng đen, được tính theo đường lấy mẫu cho trước. Ví dụ, xác định sự mở rộng của đối tượng tìm thấy.

Bám đường biên tích cực trong ảnh điện tim đồ

Ví dụ này sử dụng phương pháp bám biên tích cực, đầu vào mẫu là ảnh điện tim. Buồng tim được chỉ ra trên ảnh đầu vào.

Chương trình nguồn của ví dụ có thể tìm tại InstantVision/SIF/Samples/ActiveContour.



Bám đường biên tích cực trong ảnh điện tim đồ

Tìm kiếm nền:

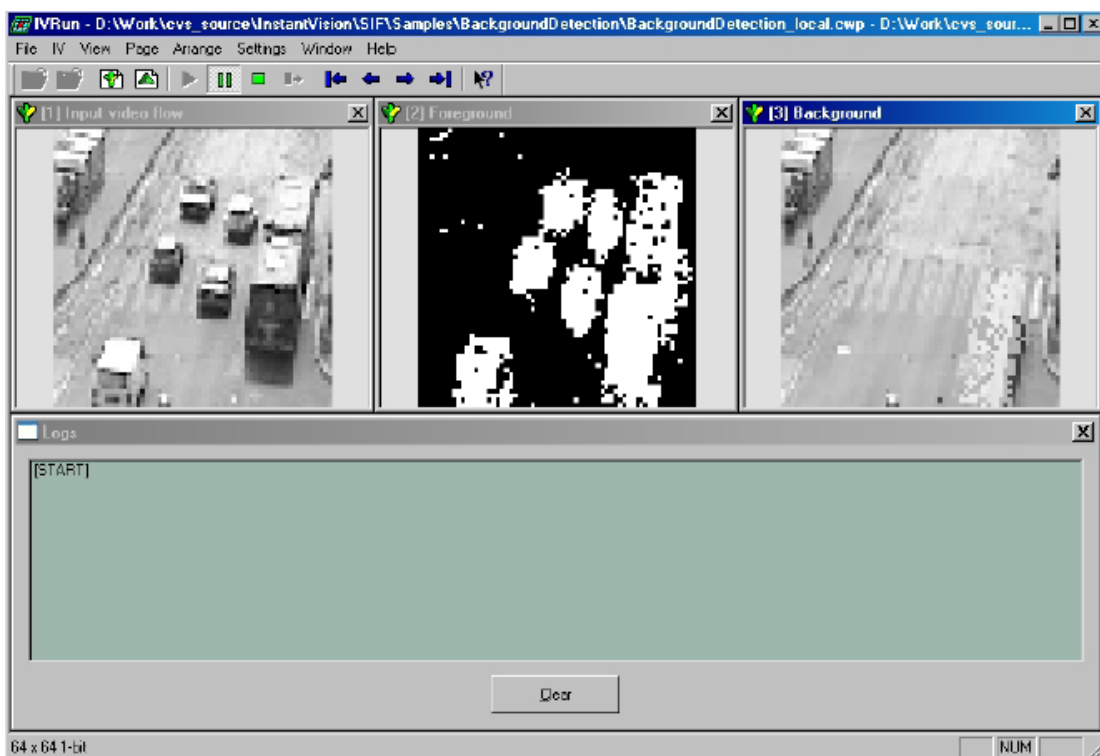
Thư viện SIF bao gồm lớp TbackgroundDetector, cung cấp một phương pháp tìm nền tổng quát của một chuỗi ảnh mức xám.

Để xác định nền từ một luồng ảnh ta có thể dựa trên phương pháp lưu và mô phỏng histogram tạm thời của mỗi điểm ảnh. Nghĩa là, ta phải lưu trữ sự thay đổi theo thời gian của các điểm để cho phép ra quyết định điểm nào thuộc nền hoặc thuộc đối tượng. Vấn đề chính xuất hiện là lưu trữ tất cả các giá trị trước của các điểm ảnh đòi hỏi cần bộ nhớ rất lớn. Một giải pháp có thể là xếp mỗi điểm ảnh trên cơ sở sử dụng vài (3-5) hàm phân bố Gaussian mô tả xác suất của các giá trị điểm thường xuyên. Mỗi hàm phân bố được xác định bởi 3 thông số: trọng số, trung tâm, và hệ số tương quan. Trung tâm xác định giá trị điểm nền tiềm năng, còn hệ số tương quan cũng như trọng số thể hiện xác suất của nó. Phân bố Gaussian thay đổi theo các giá trị của điểm vào. Điều này cho phép hệ thống mô tả nền thay đổi (ví dụ lá cây thay đổi theo gió), và để ghi nhớ phân bố cường độ của điểm ảnh trước khi thay đổi do đối tượng hoặc do nền mới. Trong trường hợp này chúng ta chỉ phải lưu giữ 3 thông số này của mỗi phân bố cho một điểm thay thế cho toàn bộ thông số trước của điểm đó.

Trong quá trình xử lý mỗi điểm, đầu tiên ta làm đồng nhất giá trị đầu vào đến tất cả các phân bố Gaussian bắt đầu với các trọng số lớn hơn. Nếu ta đồng nhất đầu vào, ta kế thừa phân bố này và loại bỏ những yếu tố khác. Nếu đầu vào phù hợp với nhiều phân bố Gaussian, thì trọng số lớn nhất sẽ được chọn là yếu tố phân bố đồng nhất. Theo như thuật toán này, trọng số Gaussian lớn nhất luôn lưu trữ nền hiện thời. Nếu một điểm đầu vào phù hợp với phân bố này, thì nó sẽ

thuộc nền, nếu không nó sẽ là đối tượng và các phân bố Gaussian khác sẽ được kế thừa. Do vậy, nếu đối tượng tĩnh xuất hiện trong chuỗi ảnh thì sau đó nó có thể trở thành nền.

-Ví dụ: Tìm kiếm nền trong bài toán quản lý giao thông



Cửa sổ thứ nhất chỉ ra luồng ảnh gốc, cửa sổ thứ hai tách đối tượng chuyển động và cửa sổ thứ ba là nền.

f. THƯ VIỆN PHÂN LOẠI THEO ĐẶC TRƯNG ẢNH (FC-Lib)

Thư viện phân loại đặc trưng, hoặc FCLib là một công cụ phần mềm tiện ích đơn giản cho bài toán phân loại hình ảnh dựa theo đặc trưng ảnh. Mục đích trong bài toán này là để xác định phân loại đối tượng thành các nhóm khác nhau được đánh nhãn bởi các số, sử dụng một số thông tin về các đặc trưng của mỗi lớp hoặc tập hợp của các mẫu đã được phân loại. Nói theo ngôn ngữ toán học là để nhận dạng một số hàm từ không gian n chiều dưới dạng tập hợp các số tự nhiên, mà đôi khi rất khó để mô tả bằng công thức. Thư viện cung cấp các phương pháp để xây dựng (bằng tay hoặc tự động), thử nghiệm và áp dụng các hệ thống phân loại, giải quyết các bài toán.

Có bốn kiểu phân loại dựa trên bốn phương pháp khác nhau: Lý thuyết mạng cộng hưởng thích nghi, tính toán Cây quyết định, Hệ thống suy luận mờ và Tìm lân cận gần nhất với các số đo khác nhau. Ngoài ra ta có thể sử dụng tổng hợp các kết quả của 4 kiểu phân loại để lựa chọn ra quyết định tối ưu nhất.

Thư viện FCLib được tổ chức như sau:

- Phân loại dựa trên Lý thuyết công hưởng thích nghi (ARTC)
- Phân loại Lân cận gần nhất (NNC)
- Phân loại dựa trên Cây quyết định (DTC)
- Phân loại dựa trên hệ suy luận mờ
- Sử dụng nhiều lớp hàm phân loại .

Các lớp hàm phân loại được thực hiện như lớp hàm C++, kế thừa từ giao diện lớp hàm CIAncestor. Lớp hàm này xác định chức năng chính như Learn(), Test(), Export() và Import().

Các ứng dụng chi tiết xem trong Tài liệu tham khảo Instance Vision.

Một số định nghĩa được sử dụng:

<i>ClassType</i>	Đánh nhãn của một nhóm đại lượng, kết quả phân loại có thể.
<i>NoClassInfo</i>	Kiểu lớp hàm được trả về khi phân loại không thuộc bất cứ kiểu lớp hàm nào đã biết.
<i>Feature</i>	Một vector n chiều mô tả một đại lượng.

Phương pháp phân lớp theo lý thuyết cộng hưởng thích nghi. (ARTC)

Phương pháp phân loại theo lý thuyết cộng hưởng thích nghi dựa trên mô hình mạng neuron ART để gắn các kiểu lớp hàm (*ClassType*) cho các vectơ đầu vào. Mạng neuron có thể được huấn luyện từng bước trong chế độ có giám sát hoặc không có giám sát. Các vectơ đầu vào để học được đưa đến hệ thống một cách lần lượt, đầu vào là kết quả học của bước trước. Trong chế độ không giám sát, các kết quả học dưới dạng tập hợp các đầu vào, trong đó, một tập hợp tương ứng với một mẫu của mạng ART. Trong chế độ học giám sát, lớp hàm *ClassType* của vectơ đầu vào phải được cho trước.

Kết quả phân loại được mã hoá theo các mức xám, trong đó, tọa độ của các điểm thể hiện các dữ liệu đầu vào, trong khi các lớp phân loại được thể hiện với các mức xám khác nhau.

Phương pháp phân loại lân cận gần nhất (NNC)

Phương pháp phân loại lân cận gần nhất tạo ra các phân lớp và tìm phân lớp gần nhất của vectơ đầu vào. Nó có thể được huấn luyện từng bước một trong chế độ có giám sát hoặc không có giám sát. Các vectơ đầu vào được đưa đến hệ thống một cách tuần tự để có thể hiệu chỉnh cấu trúc phân lớp. Trong chế độ không có giám sát, kết quả học là tập hợp các đầu vào, mỗi tập hợp tương ứng với một phân lớp được tạo ra. Trong chế độ có giám sát, *ClassType* cuối cùng của vectơ đầu vào phải cho trước, được sử dụng để gắn các lớp đích cho các tập hợp mà nó tạo ra.

Phương pháp phân loại theo Cây quyết định (DTC)

Phương pháp phân loại theo cây quyết định dựa trên phương pháp tính cây quyết định. Quá trình bắt đầu ở gốc, và tiến triển như sau: Tại mỗi nút, một quyết định cục bộ được đưa ra sử dụng một trong các giá trị của vectơ đầu vào. Kết quả của quyết định tạo ra nút con phía trái hoặc phải. Mỗi lá được gắn với một *ClassType*, và quyết định cuối cùng là *ClassType* của lá khi quá trình tính toán kết thúc.

Phương pháp phân loại dựa trên hệ suy luận mờ (FISC)

Hệ suy luận mờ (FIS) sử dụng kết quả bước trước để nhận được xác suất mà vectơ đầu vào thuộc một trong những lớp hàm *ClassType* có thể. *ClassType* có xác suất lớn nhất sẽ là kết quả của phân loại. Có một số biến với một số giá trị mờ (mô tả bởi các hàm thành viên) tương ứng với mỗi phân tử của vectơ đầu vào, và các biến mờ cho tất cả các *ClassType* có thể. Một quy tắc (cho hệ thống kiểu Sugeno bậc 0) là:

“nếu input1 là mf1 và input2 là mf2 thì clas1prob là 1 và clas2prob là 0”

Đối với một vectơ đầu vào, tất cả quy tắc đều đúng để đưa ra xác suất cuối cùng.

Phương pháp phân loại dựa trên FIS thực hiện trong lớp hàm C1Fis. Các đầu vào được thể hiện như các số, dải là tùy ý. FIS sử dụng được thực hiện trong lớp hàm Tfis, cuối cùng các hàm thành viên được thực hiện trong lớp hàm TmemFun.

Đặc điểm của FIS có thể đặt với các thông số sau đây:

<i>AlgType</i>	Kiểu hệ suy luận mờ, Sugeno hoặc Mamdani
<i>AndType</i>	Kiểu của toán hạng mờ 'and'
<i>OrType</i>	Kiểu của toán hạng mờ 'or'.
<i>DefuzzyType</i>	Phương pháp giải mờ. Trung tâm và Trung bình của Cực đại được thực hiện

Phương pháp phân loại tổng hợp

Đôi khi có thể cải thiện chất lượng của bài toán phân loại bằng việc sử dụng nhiều phương pháp phân loại khác nhau cho cùng một đối tượng (có thể có các đặc trưng khác nhau), và kết quả của phân loại được kết hợp để thu được quyết định cuối cùng. Để hỗ trợ việc này, thư viện FC-Lib cung cấp một công cụ đặc biệt, là Classifier Manager.

Classifier Manager được thực hiện trong lớp hàm C1Manager, kế thừa từ lớp hàm C1Ancestor, nên nó có thể coi như một phương pháp phân loại thứ 5.

Classifier Manager thực hiện các chức năng sau:

Quản lý các phương pháp phân loại: Các phương pháp phân loại có thể được thêm vào hệ thống, có thể được cho phép hoặc không cho phép chạy.

Quản lý các đầu vào: Classifier Manager có vector đầu vào chứa tất cả các đặc trưng cần thiết cho các phương pháp phân loại đi cùng. Từ vector đầu vào này, nó tạo ra các đầu vào chuẩn cho các phương pháp phân loại khác nhau.

Thực hiện Thử nghiệm và Học: Tại mỗi thời điểm thử nghiệm và học nó có thể gọi các hàm Test() và Lear() của các phân lớp được quản lý.

Áp dụng phương pháp tuyển lựa để kết hợp kết quả của các phương pháp phân loại theo một trong bốn thông số sau:

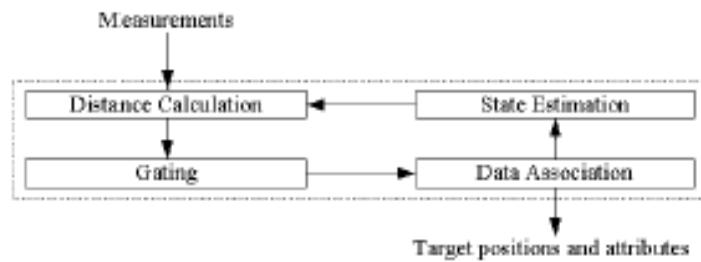
MAJORITY	ClassType có số tuyển chọn nhiều nhất sẽ là kết quả.
MAJCLASS	Giống như trên, nhưng các phân loại 'NoClassInfo' được thêm vào
ALL	Nếu tất cả các phân lớp chọn cho cùng một ClassType, thì kết quả là ClassType, nếu không thì là 'NoClassInfo'
UNIFORM	Nếu tất cả các phân loại, không chọn 'NoClassInfo' cho cùng một ClassType, thì kết quả là ClassType này, nếu không thì là 'NoClassInfo'

Lưu giữ các kết quả (cả kết quả sau khi tập hợp và kết quả của từng phép phân loại) trong ngăn xếp FIFO.

g. MODULE THƯ VIỆN BÁM ĐA MỤC TIÊU ĐỘNG (MTT-Lib)

Thư viện này cung cấp các lớp hàm, các hàm, các kiểu dữ liệu cho các ứng dụng bám đa mục tiêu chuyển động trong chuỗi ảnh. Thư viện này chứa các thuật toán xử lý tốc độ cao với bộ nhớ sử dụng ít, cho phép sử dụng trên môi trường tài nguyên hạn chế. Nó sử dụng cấu trúc dữ liệu và các lớp hàm do BaseData cung cấp.

Lớp hàm TMTTracker là lớp đối tượng trung tâm của thư viện, cung cấp các thuật toán xử lý. Hình sau chỉ ra các bước mà thuật toán TMTTracker xử lý:



Sơ đồ các bước của thuật toán MTT

Trong một ứng dụng MTT, các bước sau cần thực hiện để bám các mục tiêu:

1. *Thu thập các kết quả đo lường*

Bước này phải được thực hiện theo như toàn cảnh quá trình triển khai đã được lập kế hoạch của mỗi ứng dụng. Các nguồn thông tin về trạng thái đo lường của mục tiêu không liên quan đến bản thân các thuật toán bám.

2. *Tính toán sai lệch giữa các mục tiêu được bám hiện tại và kết quả phép đo theo như bước trên*

Trong thư viện MTT, để dễ xử lý, bước này được gộp vào bước tiếp theo. Các phương pháp nhận dạng sai lệch theo thống kê và theo hình học.

3. *Tổ chức gán tối ưu mỗi quá trình đo đến từng mục tiêu hiện thời*

Bước tổ chức này là cần thiết để nhận được chất lượng bám tốt. Nhiệm vụ này cũng là một quá trình yêu cầu nhiều thời gian tính toán. Thư viện cung cấp các ứng dụng của các thuật toán tổ chức dữ liệu khác nhau cho phép người dùng lựa chọn giữa độ chính xác bám và thời gian xử lý.

Các thuật toán cho tổ chức dữ liệu là:

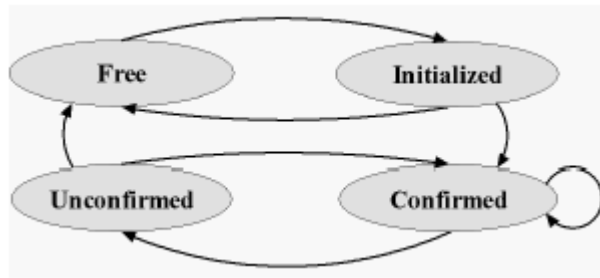
Mở rộng lân cận gần nhất(ENN) Đây là thuật toán gần tối ưu với tốc độ cao. Nó cố gắng gán mỗi phép đo đến mục tiêu gần nhất theo kiểu lần lượt mà không kiểm tra tất cả các khả năng cùng một lúc. Kết quả này cho chất lượng tốt trong trường hợp các mục tiêu không bị xáo trộn và rất gần nhau, tuy nhiên nó ít hiệu quả hơn khi nhiều đo lường tăng và/hoặc mục tiêu di chuyển nhanh hoặc xếp chồng lên nhau.

Jonker-Volgenant-Castanon (JVC) Đây là một thuật toán tối ưu toàn cục để gán dữ liệu tới lân cận gần nhất, có hiệu quả cao trong việc bám trong môi trường xáo trộn và tốc độ nhanh. Thời gian xử lý của nó tăng tỉ lệ với n^3 , trong đó n là số mục tiêu và phép đo. Đây là thuật toán thường được sử dụng trong các ứng dụng.

4. *Quản lý dấu vết: khởi tạo hoặc xoá các dấu vết*

Thư viện này sử dụng một cơ chế để quản lý dễ dàng nhất các trạng thái của dấu vết trong thời gian tồn tại của nó, như được chỉ ra trong hình vẽ dưới. Mỗi dấu vết được khởi tạo với trạng thái *Free*. Nếu các đại lượng đo không được gán với đối tượng nào, các đại lượng đo này được gán với dấu vết *Free* và chúng được chuyển đến trạng thái *Initialized*. Nếu trong khung ảnh tiếp theo, các vết *Initialized* là các phép đo được gán với đối tượng, chúng trở thành *Confirmed*, nếu không chúng bị xoá và đặt trở lại đến *Free*. Nếu một dấu vết *Confirmed* không được gán với bất cứ phép đo nào trong một ảnh, thì dấu vết trở nên

Unconfirmed. Nếu trong phép đo tiếp theo, nó vẫn không được gắn với phép đo nào, chúng trở nên *Free*, tức là dấu vết đó bị xoá.



Cơ chế bám theo dấu vết

5. Đánh giá trạng thái cho mỗi mục tiêu [14]

Các đánh giá trạng thái có thể được tính toán sử dụng các mô hình với độ phức tạp khác nhau (có thể gọi là các ước lượng trạng thái). Tất cả các mô hình tạo ra các thông số đặc biệt cho việc chỉnh định các hành vi của đối tượng được mô tả trong Tài liệu tham khảo của thư viện MTT. Các mô hình có sẵn làm tăng độ phức tạp và làm giảm chất lượng là:

Bộ lọc anpha-beta: Đánh giá vị trí tiếp theo của mục tiêu dựa trên mô hình bên trong về vị trí và vận tốc cho lần lấy mẫu tiếp theo.

Bộ lọc anpha-beta-gamma Đánh giá vị trí tiếp theo của mục tiêu dựa trên các mô hình bên trong về vị trí, vận tốc, và gia tốc cho lần lấy mẫu tiếp theo.

Bộ lọc đa mô hình tương tác (IMM) Mô hình này là sự pha trộn xác suất các mô hình theo như hành vi của các mục tiêu. Đây là mô hình phức tạp, nhưng cũng là phương pháp chính xác nhất.

Việc chỉnh các mô hình chuyển động đến các đặc tính chuyển động thực tế của các mục tiêu là rất quan trọng cho việc bám chính xác. Một trong các thông số quan trọng nhất được gọi là “chỉ số di chuyển”, là tỉ lệ của độ lệch chuẩn của sai số mô hình và độ lệch chuẩn của sai số đo lường. Nếu chỉ số di chuyển được đánh giá chính xác, các mô hình thông số tối ưu có thể tính toán dễ dàng sử dụng phương pháp *CalcFilterParams()* của lớp hàm *Ttracks*.

1.5. Một số phương pháp xử lý theo công nghệ mạng CNN

1.5.1. Thiết kế các mẫu (A, B, z) cho mạng CNN

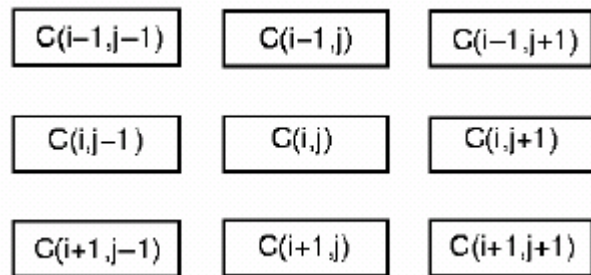
Mẫu và vai trò của mẫu [06]

Các máy tính vạn năng tương tự logic CNN-UM xử lý song song với hàng chục ngàn bộ xử lý đồng thời làm việc bởi một lệnh duy nhất đã được phát minh năm 1993 dựa trên nền tảng công nghệ mạng nơ ron tế bào CNN. Đây là một bước đột phá về chất trong lĩnh vực công cụ xử lý so với các máy tính số thực hiện hệ lệnh tuần tự. Các tế bào của mạng nơ ron (các tế bào xử lý) xử lý các mảng tín hiệu analog hoặc các mảng tín hiệu digital đồng thời cùng một lúc bằng một lệnh duy nhất cho tất cả các tế bào; điều mà các máy tính số không thực hiện được. Do tính chất động học liên tục về thời gian, bố trí không gian đồ sộ các mảng bộ xử lý nên các CNN-UM có thể thực hiện các tác vụ xử lý có yêu cầu ngặt nghèo về không-thời gian trong thời gian thực.

Các máy tính vạn năng chế tạo trên nền tảng công nghệ CNN được sử dụng đặc lực trong các lĩnh vực xử lý ảnh, quan sát đa mục tiêu di động, giải phương trình vi phân đạo hàm riêng nhiều biến và nhiều lĩnh vực khác.

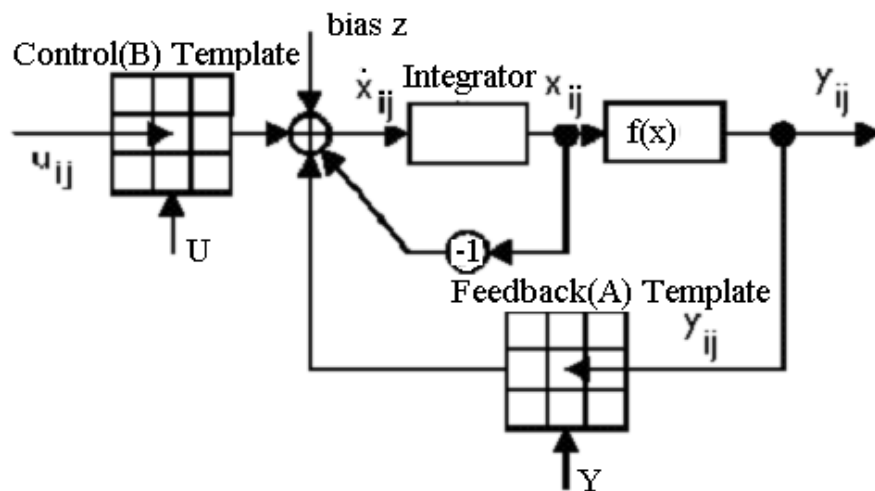
Một mạng CNN gồm nhiều tế bào (cell) đồng nhất. Mỗi tế bào C_{ij} có các tế bào láng giềng C_{kl} được định nghĩa như sau:

$$N_r(i,j) = \{C(k,l) | \max\{|k-i|, |l-j|\}\} \leq r, 1 \leq k \leq M; 1 \leq l \leq N\} \quad (55)$$



Hình 29. Mạng CNN 3x3 bán kính láng giềng $r = 1$

Hệ động lực của một tế bào trong mạng CNN được mô tả trong hình 30.



Hình 30. Hệ động lực của một tế bào CNN

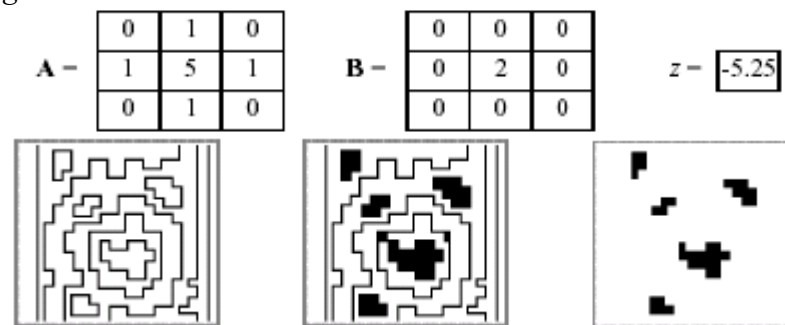
Trạng thái ổn định của mạng CNN sau thời gian quá độ ($\sim 5 \mu s$) phụ thuộc vào bộ ba ma trận trọng số liên kết (A, B, z) của mạng CNN. A, B được gọi là các toán tử hồi tiếp và các

toán tử điều khiển hoặc còn được gọi là *các mẫu vô tính* (một số tài liệu còn dùng từ *tập gien* để chỉ các toán tử này). Trong mạng CNN 2D đồng nhất, tùy theo bán kính liên kết các láng giềng của một tế bào $r = 1, r = 2$ hay $r = 3$ mà A, B sẽ là các ma trận trọng số $3 \times 3, 5 \times 5$ hoặc 7×7 . z là dòng ngưỡng của tế bào. Bộ ba trọng số liên kết (A, B, z) chính là chương trình của mạng CNN. Với mỗi yêu cầu xử lý cụ thể sẽ có một bộ (A, B, z) riêng. Phối hợp các bộ (A, B, z) khác nhau sẽ thu được kết quả mong muốn ở đầu ra của mạng CNN khi đưa các tín hiệu vào đầu vào để xử lý. Các CNN được thực hiện trên các VLSI. Trong những thời kỳ đầu, các mẫu của CNN được chế tạo cứng, như vậy rất bất tiện cho sử dụng khi cần thay đổi mẫu với các nhiệm vụ xử lý khác nhau. Về sau khi CNN-UM ra đời các mẫu được chứa trong bộ nhớ mẫu và có thể thay đổi được [09]. Ngày nay các mẫu cho CNN đã có thể được nạp và thay đổi một cách dễ dàng từ các máy tính PC sử dụng các hệ điều hành Windows 2000, XP. Mẫu là một vấn đề mấu chốt trong CNN và vẫn đang được các nhà nghiên cứu trên thế giới dành thời gian nghiên cứu nhằm tìm ra những con đường thiết kế mẫu mới, tìm ra những mẫu mới phục vụ cho các yêu cầu xử lý ngày càng đa dạng trong thực tiễn.

Thư viện mẫu CNN

Một số mẫu cơ bản đã được các nhà khoa học thuộc Viện hàn lâm khoa học Hungary soạn thảo tập hợp trong thư viện phần mềm các mẫu cho CNN (CNN Software Library 2.1) [41]. Các mẫu này có thể được sử dụng qua các hàm của các thư viện phần mềm hoạt động trên các thiết bị CNN như InstantVision chạy trên thiết bị Bi-I sử dụng chip ACE16k. Các mẫu này cũng có thể được người sử dụng nạp trực tiếp từ các file text chứa mẫu vào ACE16k và sau quá trình quá độ ta nhận được kết quả ở đầu ra. Có thể tải các mẫu từ trang web của Viện nghiên cứu máy tính và tự động hoá thuộc Viện hàn lâm khoa học Hungary ([Analogic & Neural Computing Laboratory](#), MTA-SzTAKI, Budapest, Hungary). Các mẫu này bao gồm các mẫu thực hiện các tác vụ cơ bản cho xử lý ảnh như phân ngưỡng, dò biên, thực hiện logic, hình thái học, ..ngoài ra còn có một số mẫu như mẫu dò tìm sự di chuyển trên ảnh xám, phân vùng các mẫu dệt, vải, chọn các đối tượng được đánh dấu, giải phương trình vi phân đạo hàm riêng PDE, ... Một số ví dụ về mẫu như sau:

- *Mẫu tìm vùng*

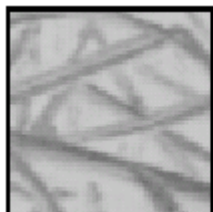


Ảnh đầu vào Khởi tạo Ảnh đầu ra

Hình 31. Mẫu tìm vùng và hiệu quả của mẫu

- *Mẫu chọn các đường nghiêng trong ảnh xám*

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} b & b & a & a & a \\ b & b & b & a & a \\ a & b & 0 & b & a \\ a & a & b & b & b \\ a & a & a & b & b \end{bmatrix} \quad z = \boxed{-1.8}$$



Ảnh đầu vào

Ảnh đầu ra

Hình 32. Mẫu tìm đường xiên và hiệu quả của mẫu

- Mẫu giải phương trình PDE Laplace

$$\mathbf{A} = \begin{bmatrix} 0 & v/h^2 & 0 \\ v/h^2 & -4*v/h^2+1/R & v/h^2 \\ 0 & v/h^2 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = \boxed{0}$$

Một số phương pháp thiết kế mẫu [2][40]

Hiện nay có nhiều phương pháp thiết kế nhằm thu được các mẫu phục vụ cho việc xử lý trên mạng CNN. Ta có thể chia các phương pháp thiết kế mẫu theo cách thức thực hiện. Cách chia này chỉ là tương đối vì một số kỹ thuật cũng có những phần giao nhau.

- *Phương pháp sử dụng các mặt nạ trong kỹ thuật số:*

Phương pháp sử dụng các mặt nạ là phương pháp quen thuộc đã được sử dụng, kiểm nghiệm trong quá trình xử lý ảnh trên máy tính số. Ví dụ các mẫu lọc trung bình (average), các mẫu erosion, dilation, ...

- *Phương pháp hệ thống dùng bảng logic*

Đây là phương pháp xác định các mẫu không liên kết cho các ứng dụng xử lý ảnh nhị phân. Bảng logic được mã hoá với các giá trị $\{0, 1\}$ tức là trắng và đen, ngoài ra chúng ta cũng còn có thể mã hoá dữ liệu nhị phân là TRUE(1), FALSE(-1) và DON'T CARE(0). Khi thiết kế các mẫu CNN để thực hiện một hàm logic $F_k(\cdot)$ đã cho, chúng ta thường sử dụng các mẫu không liên kết được mã hoá và diễn tả như sau:

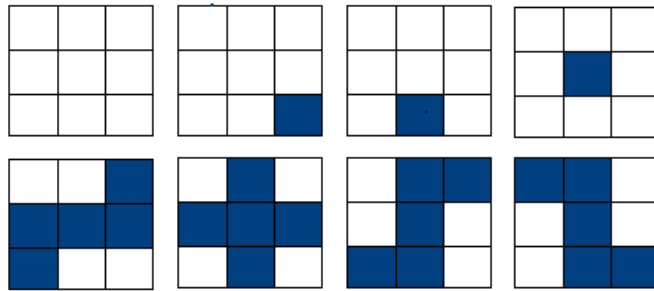
$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_{00} \geq 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}_k = \begin{bmatrix} w_{-1-1} & w_{-10} & w_{-11} \\ w_{0-1} & w_{00} & w_{01} \\ w_{1-1} & w_{10} & w_{11} \end{bmatrix} \quad \mathbf{Z} \quad (56)$$

Trong quá trình thiết kế chúng ta bắt đầu với các hàm logic 9 giá trị $F_k(u_1, u_2, \dots, u_9)$ mà giá trị khởi tạo khác không theo quy ước

$$\begin{bmatrix} u_9 & u_8 & u_7 \\ u_6 & u_5 & u_4 \\ u_3 & u_2 & u_1 \end{bmatrix} \quad (57)$$

hoặc với bảng logic. Kết quả của thiết kế là một mẫu vô tính không liên kết với các thông số $a_{0,0}, b_1, b_2, b_3, \dots, b_9, z$

Trong đó các b_i theo (56) và do mạng CNN là loại không liên kết nên $a_{0,0} = 1$. Bảng logic xây dựng gồm 2^9 trường hợp (tổ hợp của 9 giá trị u vào). Bắt đầu từ cửa số 0 tới 511. Trên hình 5 từ trái sang phải, từ trên xuống dưới là các cửa số 0, 1, 2, 16, 124, 186, 403

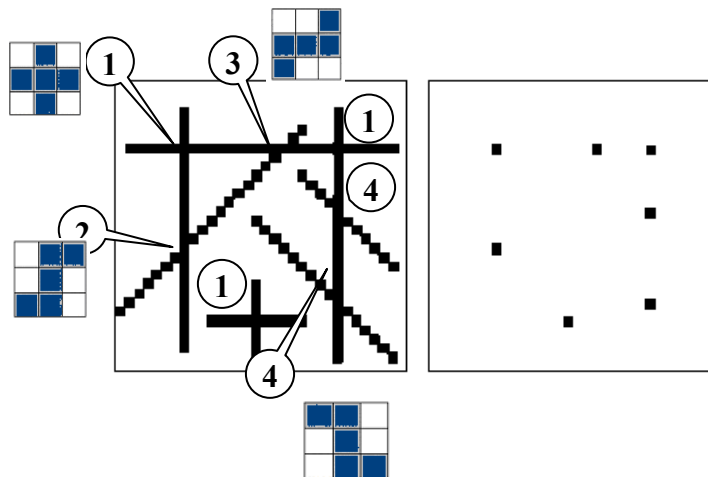


Hình 33. Một số cửa sổ bảng logic

Bảng 1 Liên quan các biến đầu vào với các mẫu

Số mẫu nhị phân	Các biến đầu vào									Hàm đầu ra $y_{ij}(\infty)$
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	
0										
1										
...										
510										
511										

Ví dụ: Giả sử chúng ta có một ảnh nhị phân với các đường có độ rộng 1 pixel. Hãy thực hiện tác vụ bằng CNN để tìm các các vị trí của các đường giao nhau 45 hoặc 90 độ.



Hình 34 a) Ảnh đầu vào

b) Ảnh đầu ra yêu cầu

Phía trái là ảnh đầu vào, phía phải là ảnh đầu ra. Hàm này là tổng của tích các minterm giá trị của $F(\cdot)$ mà tại đó giá trị =1

Chúng ta sử dụng mạng CNN không liên kết với $a_{0,0} = 1, z = -8$

Mẫu B được tìm như sau:

Xem trên hình 34 cửa sổ bảng logic cho giao nhau giữa 2 đường vuông góc (số 1) là cửa sổ 186. Giao nhau giữa đường ngang và đường chéo (số 3) là cửa sổ 124. Giữa đường đứng và các đường chéo (số 2 và số 4) là các cửa sổ 214 và 403.

Ta có:

mẫu B cho cửa sổ 186 là

$$B = \begin{bmatrix} -1 & 1 & -1 \\ 1 & 1 & 1 \\ -1 & 1 & -1 \end{bmatrix}$$

mẫu B cho cửa sổ 124 là

$$B = \begin{bmatrix} -1 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & -1 & -1 \end{bmatrix}$$

mẫu B cho cửa sổ 214 là

$$B = \begin{bmatrix} -1 & 1 & 1 \\ -1 & 1 & -1 \\ 1 & 1 & -1 \end{bmatrix}$$

mẫu B cho cửa sổ 403 là

$$B = \begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix}$$

Các mẫu A và z là

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad Z = -8$$

Các cặp mẫu sẽ được thực hiện trên ảnh đầu vào 6a. Kết quả 6b là tổng hợp của các thực hiện thành phần.

- Phương pháp thiết kế mẫu dùng cách giải tập bất phương trình các luật cục bộ

Để hiểu rõ các luật cục bộ trong đặc tính động học của mạng CNN, chúng ta phân tích một mẫu thực hiện một tác vụ rất quen thuộc là mẫu dò biên.

Mẫu dò biên nhị phân có dạng như sau

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad z = \boxed{-1}$$

Đầu vào $U(t) = P$
 Điều kiện khởi tạo $x(0) = \text{Tùy ý}$

Điều kiện biên: Kiểu cố định $u_{ij} = 0, y_{ij} = 0$ cho các tế bào ảo ký hiệu là $[U]=[Y]=0$

Đầu ra: $Y(t) \Rightarrow Y(\infty) = \text{Ảnh nhị phân đã dò biên P}$

Các luật cục bộ:

Bảng 2 - Luật cục bộ trong tác vụ dò biên

	Đầu vào tĩnh u_{ij}	Đầu ra trạng thái ổn định $y_{ij}(\infty)$
1.	Điểm trắng	→ trắng. Không phụ thuộc vào các láng giềng
2.	Điểm đen	→ trắng nếu tất cả các láng giềng gần nhất là đen
3.	Điểm đen	→ đen, nếu ít nhất một láng giềng gần nhất là trắng
4.	Điểm đen, xám hoặc điểm trắng	→ xám nếu các láng giềng gần nhất là xám

Phương trình trạng thái (1a) có thể được viết thành:

$$\begin{aligned} \dot{x}_{ij} &= g(x_{ij}) + w_{ij} = h_{ij}(x_{ij}, w_{ij}) \\ y_{ij} &= f(x_{ij}) \end{aligned} \quad (58)$$

trong đó $g(x_{ij}) = -x_{ij}$ và

$$\begin{aligned} w_{ij} &= -1 + 8u_{ij} - u_{j+1,j-1} - u_{i+1,j} - u_{i+1,j+1} - u_{i,j-1} - u_{i,j+1} - u_{i-1,j-1} - u_{i-1,j} - u_{i-1,j+1} \\ &= -1 + 8u_{ij} - \sum_{\substack{kl \in S_1(i,j) \\ kl \neq ij}} u_{kl} \end{aligned}$$

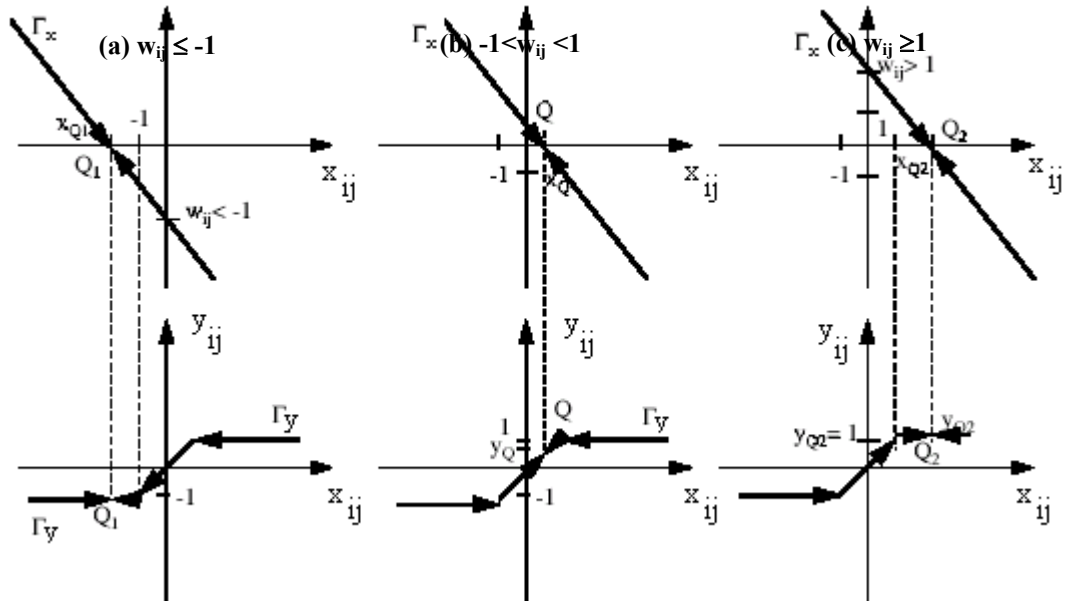
Với bất kỳ điều kiện khởi tạo nào $x_{ij}(0) \in R$ với bất kỳ hằng số đầu vào nào u_{ij} mạch điện CNN sẽ ổn định hoàn toàn ở các trạng thái mà không phụ thuộc vào điều kiện khởi tạo và có xu hướng đi về điểm cân bằng của phương trình (1.5-4)

Khi $y_{ij}(\infty) = \lim_{t \rightarrow \infty} y_{ij}(t) \in R$ ngoài ra nếu $u_{ij} \in \{-1, 1\}$ thì $y(\infty) \in \{-1, 1\}$

$$y(\infty) = \begin{cases} w_{ij} & \text{nếu } |w_{ij}| < 1 \\ 1 & \text{nếu } w_{ij} \geq 1 \\ -1 & \text{nếu } w_{ij} \leq -1 \end{cases} \quad (59)$$

Xem trên quỹ đạo DP (driving-point) đường quỹ tích Γ của $h_{ij}(x_{ij}, w_{ij})$ cắt trục ngang ở điểm Q.

$$\dot{x}_{ij} = h_{ij}(x_{ij}, w_{ij}) \quad \dot{x}_{ij} = h_{ij}(x_{ij}, w_{ij}) \quad \dot{x}_{ij} = h_{ij}(x_{ij}, w_{ij})$$



Hình 35. Các trạng thái động học của mẫu dò biên

Các luật cục bộ có thể phát biểu thành:

- + Luật 1: Nếu $u_{ij} = -1$ thì $y_{ij}(\infty) = -1$ không phụ thuộc vào $u_{kl} \in \{-1, 1\}$ $k, l \in S_1(i, j)$
- + Luật 2: Nếu $u_{ij} = 1$ và $u_{kl} = 1$ với mọi $kl \in S_1(i, j)$ thì $y_{ij}(\infty) = -1$
- + Luật 3: Nếu $u_{ij} = 1$ và $u_{\alpha\beta} = -1$ với một số $C(\alpha, \beta) \in S_1(i, j)$ thì $y_{ij}(\infty) = 1$
- + Luật 4: Nếu $u_{ij} \in [-1, 1]$ và $u_{kl} \in [-1, 1]$ $k, l \in S_1(i, j)$ thì $y_{ij}(\infty) \in [-1, 1]$

Qua ví dụ phân tích một mẫu trên chúng ta đã hiểu thế nào là luật cục bộ. Để đảm bảo các mẫu cho phép CNN hoạt động ổn định phải xây dựng được các tập luật đầy đủ và đúng đắn qua các bất phương trình sau đó giải các bất phương trình này để tìm A, B và z.

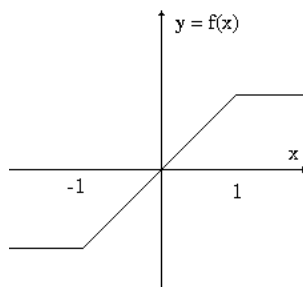
• *Phương pháp học*

Ta biết mỗi một tế bào là một neuron. Trong mạng CNN các mẫu là tập các trọng số liên kết giữa các neuron. Vậy thì chúng ta cũng có thể áp dụng các phương pháp học để tìm bộ các trọng số (các mẫu này). Đã có nhiều mô hình học cho mạng CNN. Ta khảo sát mô hình xác định mẫu cho mạng CNN nhị phân. Nội dung của phương pháp này như sau:

Phương trình động học trạng thái của tế bào CNN có thể viết thành

$$\frac{dv_{xij}}{dt} + v_{xij} = a^T v_y + b^T v_u + I \quad \text{với } 1 \leq i \leq M; 1 \leq j \leq N \quad (60)$$

phương trình đầu ra
$$v_{yij}(t) = \frac{1}{2} (|v_{xij}(t) + 1| - |v_{xij}(t) - 1|) \quad (61)$$



Hình 36: Hàm đầu ra (Piecewise - linear output function)

$a \in R^9$ là toán tử hồi tiếp $b \in R^9$ là toán tử điều khiển $I \in R$ là nguồn dòng ngưỡng, $v_{xij} \in R$ là biến trạng thái của $C(i,j)$, $v_u \in R^9$ $v_y \in R^9$ là các vectơ đầu vào và đầu ra được đặt vào $C(i,j)$.

Giả định rằng đầu ra $v_{yij}(t)$ chỉ có thể lấy các giá trị nhị phân +1 hoặc -1, với $t \rightarrow \infty$. Nếu các $v_{yij}(\infty)$ là tập các đầu ra nhận được và theo (61) ta có

$$a^T v_y + b^T v_u + I \geq 1 \quad \text{nếu } v_{xij} = 1 \quad (62)$$

$$a^T v_y + b^T v_u + I \leq -1 \quad \text{nếu } v_{xij} = -1 \quad (63)$$

Từ (62) và (63) thu được công thức tổng quát

$$\pm (a^T v_y + b^T v_u + I \mp 1) \geq 0 \quad (64)$$

Để thu được các mẫu A, B đối xứng cần thêm một số đẳng thức bất buộc

$$\begin{aligned} a_1 = a_3 = a_7 = a_9 & \quad a_2 = a_4 = a_6 = a_8 \quad \text{và} \\ b_1 = b_3 = b_7 = b_9 & \quad b_2 = b_4 = b_6 = b_8 \end{aligned} \quad (65)$$

với các chỉ số mẫu như hình 37

a ₁	a ₂	a ₃
a ₄	a ₅	a ₆
a ₇	a ₈	a ₉

Hình 37: Chỉ số mẫu với mẫu A

Lấy $x^T = [a^T, b^T, I]$ là một vectơ 19 chiều. Thuật toán học dựa trên các điều kiện (64) và (65) nhằm thu được các mẫu như sau

Thuật toán:

Bước 0: Chọn $x \in R^{19}$ tùy ý

Bước 1: For i = 1 to N

For j = 1 to M

Nếu ($v_{yij} = 1$) và ($[v_y^T, v_u^T, I].x - 1 < 0$)

$$\text{thì } x := x - 2 \frac{[v_y^T, v_u^T, 1].x - 1}{\|v_y\|^2 + \|v_u\|^2 + 1} \begin{bmatrix} v_y \\ v_u \\ 1 \end{bmatrix}$$

else

Nếu ($v_{yij} = -1$) và ($-[v_y^T, v_u^T, 1].x - 1 < 0$)

$$\text{thì } x := x - 2 \frac{[v_y^T, v_u^T, 1].x + 1}{\|v_y\|^2 + \|v_u\|^2 + 1} \begin{bmatrix} v_y \\ v_u \\ 1 \end{bmatrix}$$

nếu $x_5 < 1$ thì $x_5 := 2 - x_5$

Bước 2: For i = 1 to 12

$$\text{Nếu } x \text{ vi phạm điều kiện (11) thì } x := x - \frac{e_i^T x e_i}{2}$$

Bước 3: Nếu $x \in S$ thì kết thúc

nếu không quay lại bước 1

(S là tập không rỗng các lời giải đảm bảo cả các điều kiện (64) và(65). Thuật toán này đã được chứng minh là luôn hội tụ vì vậy chúng ta luôn thu được tập mẫu sau một số lần lặp.

Thời gian qua trong quá trình thiết kế các mẫu, hầu hết tất cả các phương pháp huấn luyện mạng nơron kinh điển đã được đưa vào với mạng CNN. Nhưng có 3 vấn đề quan trọng cần lưu ý với các kết quả và kỹ thuật này:

- + Trước hết học là trên cơ sở một cặp đầu vào và đầu ra yêu cầu, và trong khi học một thủ tục tốt hơn và kết quả tốt hơn thì được cho là sinh ra mẫu tốt hơn. Nhưng trong nhiều trường hợp (nhất là sinh ra kiểu mẫu giống như CCD , mẫu có thể làm việc hoặc không làm việc và không phải cứ thủ tục tốt hơn thì kéo theo mẫu tốt hơn. Điều này khiến người ta có thể thay đổi chiến lược dùng phương pháp cưỡng ép mà không dùng phương pháp học.
 - + Vấn đề thứ hai là một số trường hợp mẫu học không hội tụ.
 - + Vấn đề thứ ba một số phương pháp học đề xuất đã được kiểm tra và chứng minh thì có hiệu lực tới các nhóm mẫu. Nhưng kết quả có thể nhận được trực tiếp từ các diễn tả tác động chính xác rõ ràng thì không cần thiết phải sử dụng phương pháp học. Tuy nhiên có một số trường hợp mẫu học là phương pháp thiết kế rất quan trọng khi mà đầu ra yêu cầu tồn tại một cách rõ ràng, việc thiết kế mẫu trực tiếp không thực hiện được ví dụ như trong bài toán việc phân loại các mẫu dệt.
- Thiết kế mẫu dùng giải thuật di truyền [33]

Thuật giải di truyền (*Genetic algorithms- GA*) có đặc điểm thích hợp với những bài toán dạng tối ưu. Bản chất của giải thuật di truyền là một giải thuật tìm kiếm ngẫu nhiên có định hướng (hàm thích nghi). Thuật gen có khả năng kiểm tra hết mọi trường hợp có thể của bài toán, nhưng đó là một sự kiểm tra ngẫu nhiên và có định hướng; khác với phương pháp vét cạn tất cả các trường hợp. Có thể xem thuật gen như là một phương pháp khám phá. Thuật giải di truyền không quan tâm đến nội dung của bài toán mà chỉ cần quan tâm đến vấn đề cần giải quyết của bài toán. Vì GA là một giải thuật ngẫu nhiên nên nếu có thể xác định được hàm phân bố ngẫu nhiên của bài toán thì việc giải quyết bài toán sẽ dễ dàng hơn. Các giải thuật leo đồi, luyện thép cũng đã được áp dụng để tìm kiếm mẫu. Tuy nhiên giải thuật di truyền không phải khi nào cũng cho kết quả tối ưu và giá thành tính toán đắt (Số lượng phép tính lớn). Mặc dù vậy do là một kỹ thuật tìm kiếm giải pháp dựa trên việc mô phỏng quá trình tiến hoá của tự nhiên nên hiện nay kỹ thuật này vẫn được các nhà nghiên cứu CNN áp dụng để tìm kiếm, thiết kế mẫu. Trong hội nghị quốc tế về CNN 8-2006 tại Thổ Nhĩ Kỳ vừa qua kỹ thuật này vẫn được ứng dụng trong một số báo cáo.

- Kỹ thuật thiết kế fuzzy:

Lý thuyết tập mờ và logic mờ (fuzzy sets and fuzzy logic) do giáo sư Lofti Zadeh công bố năm 1965. Logic mờ được thiết kế để làm việc với các sự kiện không chính xác. Mặc dù không được các hầu hết cộng đồng nghiên cứu ủng hộ trong phát triển lý thuyết, logic mờ vẫn được áp dụng và có nhiều thành công trong thực tiễn như trong hệ thống điện ô tô, máy điều hoà nhiệt độ, thang máy, máy giặt, máy rửa bát, các bộ lọc ngôn ngữ, xử lý ảnh số, ...

Logic mờ cũng được áp dụng trong thiết kế các mẫu phục vụ cho các tác vụ CNN.

- Khai thác các mẫu sinh học:

Là phương pháp sử dụng một số kiểu hình trạng nơron (neuromorphic) của một cá thể sống, tiêu biểu như hệ thống thần kinh thị giác của động vật có xương sống và thể hiện các mẫu này bằng VLSI.

Kết luận

Ngoài các mẫu đã được giới thiệu trong thư viện mẫu, với mỗi bài toán cụ thể của thực tế cần có một bộ mẫu riêng hoặc chuỗi bộ mẫu khi sử dụng công nghệ CNN. Đã có nhiều phương pháp thiết kế mẫu khác nhau nhằm thu được các mẫu mạnh và ổn định khi xử lý trên các máy tính vạn năng tương tự - logic CNN-UM. Những phương pháp như dùng bảng logic, giải các bất phương trình với các luật cục bộ cho chúng ta các kết quả ổn định. Các phương pháp học, lập trình tiến hoá, ứng dụng logic mờ cũng được áp dụng, tuy nhiên với mỗi phương pháp cũng có một vài khiếm khuyết như đã trình bày ở phần 2. Việc nghiên cứu mẫu và thiết kế mẫu sẽ ảnh hưởng đến mọi nhánh nghiên cứu ứng dụng công nghệ CNN. Trong tương lai sẽ còn cần rất nhiều nỗ lực của các nhà nghiên cứu về CNN với các yêu cầu về mẫu và phương pháp thiết kế mẫu đặc biệt là với các CNN nhiều chiều, nhiều lớp.

1.5.2. Mô hình hóa phương trình đạo hàm riêng sử dụng mạng CNN

Trong nhiều bài toán khoa học các đại lượng biến thiên phức tạp theo nhiều tham số không gian, thời gian, các điều kiện ngoại cảnh. Để giải quyết các bài toán trên thường đưa đến việc giải phương trình vi phân, thậm chí là phương trình vi phân đạo hàm riêng. Việc nghiên cứu ứng dụng máy tính vào giải phương trình đạo hàm riêng là cần thiết. Các máy tính thông thường hiện nay có thể giải được nhưng với tốc độ chậm, một số trường hợp không đáp ứng được với ứng dụng trong thời gian thực.

Việc áp dụng công nghệ mạng nơ ron tế bào CNN vào mô hình hóa để giải phương trình đạo hàm riêng với tốc độ cao là cần thiết và có nhiều triển vọng trong tương lai đáp ứng cho các bài toán trong thời gian thực [28],[29],[37],[3]. Phần này đề xuất đến việc nghiên cứu một số dạng phương trình vi phân đạo hàm riêng để có thể giải bằng công nghệ CNN.

Phương trình đạo hàm riêng PDE (Partial Differential Equation) là phương trình có chứa đạo hàm riêng của hai hay nhiều hơn hai biến của hàm phải tìm. Ví dụ:

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} - \frac{\partial u}{\partial z} = 0 \quad (66)$$

$$\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = u \quad (67)$$

(66) và (67) là các phương trình đạo hàm riêng của hàm chưa biết là $u(x,y,z)$

Trong phân tích các PDE ta luôn luôn có thể rời rạc theo không gian và thu được tập các phương trình vi phân thường. Nghĩa là từ PDE chúng ta có thể phân rã thành một ma trận các phần tử (mỗi phần tử có thể lại là một hệ thống) nhỏ và rời rạc nhưng có tương tác với nhau thông qua các tham số.

Để minh họa cách tiếp cận CNN trong việc giải phương trình PDE ta xét một số ví dụ về giải phương trình đạo hàm riêng sau:

Phương trình mô tả hàm nhiệt trong không gian hai chiều:

$$\frac{\partial u(x, y, t)}{\partial t} = c \nabla^2 u(x, y, t) \quad (68)$$

Với ∇^2 là toán tử Laplace của cường độ $u(x,y,t)$ có đạo hàm riêng có dạng:

$$\nabla^2 u_i = \frac{\partial^2 u_i}{dx^2} + \frac{\partial^2 u_i}{dy^2} \quad \text{với } i=1,2,..m; \quad c \text{ là hằng số.}$$

Với những phương trình như trên ta dùng phương pháp rời rạc theo không gian chuyển hệ PDE thành tập các ODE mà với các hàm trạng thái là một tập mẫu thích hợp của CNN. Ta xây dựng lưới không gian cách đều nhau có bước là h theo cả hai hướng x, y . Chúng ta ánh xạ $u(x,y,t)$ thành các lớp CNN mà mỗi điểm i,j có điện áp trạng thái của mỗi cell là $v_{xij}(t)$ tương ứng với $u(ih,jh,t)$ của hàm gốc. Ta được một cách tổng quát các phương trình PDE dựa trên khai triển Taylor của $u(x,y,t)$. Đạo hàm riêng cấp hai được tính như sau:

$$\frac{\partial^2 u}{\partial x^2} \sim \frac{1}{h^2} [u(x+h, y) - u(x, y) - (u(x, y) - u(x-h, y))]$$

$$= \frac{1}{h^2} [u_{i+1,j} - 2u_{i,j} + u_{i-1,j}] \quad (69)$$

Trong đó h là khoảng cách lưới đồng nhất. Sử dụng phương pháp xấp xỉ ta thiết kế được mẫu CNN như sau:

a. Một chiều $\left[\frac{1}{h^2}, -\frac{4}{h^2}, \frac{1}{h^2} \right]$

b. Hai chiều $A = \begin{bmatrix} 0 & \frac{1}{h^2} & 0 \\ \frac{1}{h^2} & -\frac{4}{h^2} + \frac{1}{R} & \frac{1}{h^2} \\ 0 & \frac{1}{h^2} & 0 \end{bmatrix}$, $B = 0$, $I = 0$ (70)

mẫu này đóng vai trò như toán tử Laplace. Bằng việc mắc thêm một tụ song song với đầu ra của tế bào phương trình trên được đưa vào hệ CNN

Sử dụng cách tiếp cận này phương trình đạo hàm riêng được giải một cách có hiệu quả với thời gian chấp nhận được. Sau đây là một số ví dụ trong việc mô hình hóa một số hệ PDE bằng mạng nơ ron tế bào:

Phương trình Burger

Phương trình phi tuyến:

$$\frac{\partial u(x,t)}{\partial t} = \frac{1}{R} \frac{\partial^2 u(x,t)}{\partial x^2} - u(x,t) \frac{\partial u(x,t)}{\partial x} + F(x,t) \quad (71)$$

gọi là phương trình Burger mô tả ý nghĩa của mật độ $u(x,t)$ của các phân tử khí chuyển động dọc theo trục tọa độ x như là một hàm về thời gian t . Giả sử là vận tốc của các phân tử tăng lên tuyến tính với mật độ phân tử. Giải phương trình với trường hợp đồng nhất ($F(x,t) = 0$). Ta có:

$$u(x,t) = \frac{\int_{-\infty}^{+\infty} (x-y) \exp\left\{ \frac{R}{4t} (x-y)^2 - \frac{R}{2} \int_0^y u(y',0) dy' \right\} dy}{t \int_{-\infty}^{+\infty} \exp\left\{ \frac{R}{4t} (x-y)^2 - \frac{R}{2} \int_0^y u(y',0) dy' \right\} dy}, \quad \int_{-\infty}^{+\infty} u(y,0) dy < \infty \quad (72)$$

Thay thế các đạo hàm riêng với: $u_i(t) \equiv u(x_i,t)$, $F(x_i,t)$ và $\Delta x = x_{i+1} - x_i$; phương trình (71) có thể xấp xỉ bằng một tập các hệ phương trình vi phân thường ODE:

$$\frac{du_i}{dt} = \frac{1}{R} \frac{u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)}{(\Delta x)^2} - \frac{u_i(t)[u_{i+1}(t) - u_{i-1}(t)]}{2\Delta x} + F_i(t) \quad (73)$$

với $i=1 \dots N-1$

Chúng ta đặt $u_0(t) = u_M(t) = 0$ và giả sử rằng $u(x,t)$ coi như bằng 0 ngoài đoạn $[x_0, x_M]$. So với các chỉ số của phương trình (73) với hàm trạng thái của CNN phi tuyến chúng ta tìm được mẫu sau:

$$\begin{aligned}
A_{LIN} &= \left(\frac{1}{R(\Delta x)^2}, \frac{1}{R_c} - \frac{2}{R(\Delta x)^2}, \frac{1}{R(\Delta x)^2} \right), [u_k] \\
A_{NLIN} &= \left(\frac{1}{2(\Delta x)}, 0, -\frac{1}{2(\Delta x)} \right), [u_i u_k] \\
B &= (0, 1, 0), [F_k]
\end{aligned} \tag{74}$$

Mẫu hồi tiếp $\tilde{A} = A_{LIN} + A_{NLIN}$ được tách thành hai thành phần tuyến tính A_{LIN} và phi tuyến A_{NLIN} . Giải pháp của phương trình Burger có thể tính toán với các giá trị khác nhau của R , Δx và các sơ kiện khác nhau $u(x,0)$.

Phương trình Sine-Gordon.

$$\frac{\partial^2 \varphi}{\partial t^2} + \sin \varphi = \frac{\partial^2 \varphi}{\partial x^2} \tag{75}$$

Để giải phương trình này ta chuyển PDE thành hệ các ODE mà được xác định bởi các hàm trạng thái của CNN với các mẫu tương ứng. Việc rời rạc hoá không gian được thực hiện với các bước $h = \Delta x$. Chúng ta ánh xạ $\varphi(x,t)$ thành các lớp CNN mà điện áp trạng thái của mỗi cell $x_{ij}(t)$ tại mỗi điểm i,j trong lưới. Phương trình đạo hàm riêng cấp hai được biểu diễn:

$$\frac{\partial^2 \varphi}{\partial x^2} \sim \frac{1}{h^2} [\varphi(x+h,t) - \varphi(x,t) - (\varphi(x,t) - \varphi(x-h,t))] = \frac{1}{h^2} [\varphi_{i+1,j} - 2\varphi_{i,j} + \varphi_{i-1,j}] \tag{76}$$

Giả sử rằng hàm vào ($u_{ij}(t) \equiv 0$) như vậy ở đây ta chỉ xét với CNN có đầu vào bằng 0 như một miền riêng hữu hạn của hệ CNN tổng quát. Trong việc khảo sát nghiên cứu các điều kiện của bài toán ta cần giả sử rằng hệ CNN có kích thước xác định. Các trường hợp khác điều kiện biên được thay bằng mô hình vũ trụ ($M \rightarrow \infty$). Chúng ta giả thiết rằng hệ trên đồng nhất về không gian hay gọi là đẳng hướng có nghĩa là hệ số mẫu a_{ijkl} là độc lập với i,j và vì vậy sẽ chỉ tùy theo k,l .

Chúng ta hãy xem xét hệ CNN một chiều khi mà các cell trong CNN chỉ gồm các tụ tuyến tính mắc song song với cuộn cảm phi tuyến mô tả bởi $i_i = f(\varphi_i)$. Các cell này toạ thành từng cặp với nhau bởi các tụ tuyến tính và cuộn cảm L . Trong phạm vi hình trạng mạch CNN chúng ta có thể xác định các phân tử tương ứng sau:

$$\text{a. Cell CNN động} \quad \frac{dv_i}{dt} = -\frac{1}{C} [f(v_i) - I_i] \tag{77}$$

$$\frac{d\varphi_i}{dt} = v_i \tag{78}$$

$$\text{b. Luật tiếp hợp CNN} \quad I_i = i_{L_i} - i_{L_{i-1}} = \frac{1}{L} (\varphi_{i-1} - 2\varphi_i + \varphi_{i+1}) \tag{79}$$

Khi $\varphi_i(t) = \int_{-\infty}^t v_i(\tau) d\tau$ là thông lượng liên kết tại node j . Ta thấy rằng luật tiếp hợp (79) là toán tử Laplace rời rạc một chiều của thông lượng liên kết φ_i . Với cấu trúc CNN trong miền riêng này ta có mẫu như sau:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Với toán tử * nguồn điều khiển tiếp hợp được xác định:

$$a_i * f(v_{i+1}, v_i) = a_i \int_{-\infty}^t [v_i(\tau) - v_{i+1}(\tau)] d\tau = a_i (\varphi_i - \varphi_{i+1})$$

Nếu thay (78) và (79) vào (77) chúng ta nhận được

$$C \frac{d^2 \varphi_i}{dt^2} + f(\varphi_i) = \frac{1}{L} \nabla^2 \varphi_i \quad \text{với } j=0,1,2,\dots,k-1,k,k+1\dots \quad (80)$$

Giả sử ta chọn điện cảm phi tuyến là hàm Josephson mô tả bởi $i_t = f(\varphi_i) = \sin(\varphi_i)$ và chọn đơn giản $L=C=1$, ta sẽ nhận được phương trình Sine-Gordon.

Mô hình CNN với phương trình FitzHugh-Nagumo

Mô hình toán học được sử dụng rộng rãi nhất của việc kích thích và truyền xung lực trong hệ thần kinh là phương trình FitzHugh-Nagumo. Người ta đã xác định được phương trình này - việc truyền xung tích cực có thể hợp nhất dưới ảnh hưởng của hiện tượng CNN truyền tương tác một chiều khi mà các cell ở tình trạng suy giảm trong máy phát dao động của L.O. Chua. Trong phần này chúng ta tập trung vào vấn đề CNN truyền tương tác với luật tiếp hợp tuyến tính mà xấp xỉ với toán tử Laplace không gian.

Vùng CNN cục bộ trong hệ CNN hình thành từng cặp bởi một lớp các trở tuyến tính tạo ra một thiết bị đồng bộ, các hoạt động của chúng sinh ra hiện tượng sóng. Trong trường hợp này vì phương trình FitzHugh-Nagumo là dạng đơn giản hoá của phương trình Hodgkin - Huxley cổ điển mô tả việc truyền tín hiệu của hệ thần kinh. Chúng ta có thể chứng minh rằng việc chọn chính xác tham số mạch hệ CNN sẽ mô tả tổng quát đa dạng hơn việc truyền tín hiệu thần kinh.

Xét dạng phương trình FitzHugh-Nagumo:

$$\begin{aligned} u_t &= u(u-a)(1-u) - \omega + u_{xx} \\ \omega_t &= \varepsilon(u-b\omega) \end{aligned} \quad (81)$$

Khi u_t là đạo hàm riêng bậc 1 của $u(t,x)$ với t , u_{xx} là đạo hàm riêng cấp 2 của u với x , ω_t đạo hàm riêng cấp 1 của $\omega(t,x)$ với $t, b \geq 0; 0 < a < 1/2; 0 < \varepsilon \ll 1$; u là thế năng của hệ thần kinh ω là biến trạng thái. Trong hàm này trạng thái ổn định $u = \omega = 0$ diễn tả cho trạng thái nghỉ của tế bào thần kinh. Với ε nhỏ, ω biến thiên chậm so với u và trong thời gian quá độ chúng ta có thể giả sử rằng ω thay đổi không đáng kể nên coi gần đúng $\omega = 0$. Vì vậy:

$$u_t = u(u-a)(1-u) + u_{xx}$$

Ở đây $u = 0$ tương ứng với trạng thái nghỉ và $u=1$ là trạng thái kích hoạt của thần kinh. Phương trình này đã chứng minh rằng $u = 0$ và $u = 1$ là ổn định trái lại $u = a$ là không ổn định vì vậy đây là hiện tượng ngưỡng. Mô hình với $\omega = 0$ có thể dự đoán rằng sự kích thích cục bộ xác định có thể đủ kích hoạt mặt trước của sóng như là tế bào thần kinh trong trạng thái nghỉ trước khi chuyển trạng thái kích hoạt sau đó. Do vậy thông tin được truyền đi dọc theo hệ thần kinh. Để mô hình truyền tín hiệu thần kinh thành hiện thực cần phải có một cơ cấu trở về từ trạng thái nghỉ để tế bào thần kinh được kích hoạt trở lại và đây là tác dụng của biến ω

Những cơ hội và thách thức của CNN

Như ta đã nghiên cứu hệ CNN có những khả năng ứng dụng to lớn trong việc tính toán với tốc độ cao. Công nghệ này đáp ứng với những lớp bài toán của xử lý thời gian thực không gian vào có tham số lớn. Ngày nay với sự phát triển nhanh chóng của công nghệ vi mạch hệ CNN có khả năng mở rộng số lượng các cell trong hệ thống, đồng thời mở rộng không gian (biên) bài toán. Với cấu trúc phần cứng của cell ta còn có thể bổ xung các phần tử mạch khi cần thiết để tăng khả năng giải quyết bài toán trong CNN.

Việc nghiên cứu ứng dụng công nghệ CNN ở Việt Nam còn chưa nhiều. Nhiều tính năng, đặc điểm, cũng như các bài toán ứng của CNN chưa được phát hiện và triển khai. Những dạng phương trình đạo hàm riêng đưa ra ở đây chỉ có tính tổng hợp và giới thiệu. Chúng cần được nghiên cứu tỉ mỉ cài đặt chạy thử so sánh với cách giải trên máy tính thông thường mới đánh giá sự đúng đắn của nó. Từ đó có thể phát triển ứng dụng trong một nhiệm vụ tính toán cụ thể. Đây là cơ hội cho lực lượng cán bộ trẻ nghiên cứu và phát triển.

Tuy vậy, việc nghiên cứu mô hình hóa các bài toán phương trình đạo hàm riêng trên CNN cũng có nhiều thách thức. Đây là các bài toán ứng dụng nên việc khảo sát đưa ra bài toán hoàn chỉnh với các ràng buộc, điều kiện biên đúng đắn chính xác không dễ dàng.

Việc thiết kế các mẫu (templates) đưa vào CNN cũng là vấn đề cần nghiên cứu, bắt buộc ta phải hiểu tỉ mỉ cấu trúc mạch của các hệ CNN (một lớp, nhiều lớp...) vì trong khi đưa các mẫu vào có thể phải thêm vào các phần tử mạch tuyến tính, hay phi tuyến. Việc can thiệp vào cấu trúc mạch điện của CNN cần có những thử nghiệm phức tạp. Điều này khác hẳn các bài toán triển khai trên máy tính thông thường.

1.5.3. Mô hình mắt nhân tạo sử dụng mạng CNN

Giới thiệu chung

Sự phát triển của công nghệ thông tin và vi điện tử ngày nay đã tạo điều kiện cho nhiều lĩnh vực nghiên cứu mới ra đời mà ứng dụng của chúng đã và đang hơn bao giờ hết làm cho khoa học công nghệ trở nên gần gũi với con người. Trong số đó, các nghiên cứu về Tin sinh học (Bioinformatics) nói chung và thị giác nhân tạo nói riêng là một lĩnh vực còn mới mẻ và đang được các nhà nghiên cứu trên thế giới quan tâm. Nghiên cứu về thị giác sinh học, võng mạc nhân tạo, không chỉ nhằm mục đích lâu dài là chế tạo được một cấu trúc thị giác phỏng sinh học giúp người khiếm thị nhìn thấy được, mà trước mắt có thể ứng dụng trực tiếp trong nhiều lĩnh vực của đời sống, trong các dây chuyền sản xuất công nghiệp, trong các hệ thống an ninh, quốc phòng... Phần này sẽ trình bày tóm tắt một số kết quả nghiên cứu về thị giác nhân tạo trên thế giới và một số kết quả nghiên cứu hứa hẹn nhiều triển vọng trong nghiên cứu thị giác nhân tạo ứng dụng công nghệ tính toán mạng nơ-ron/phi tuyến tế bào CNN.

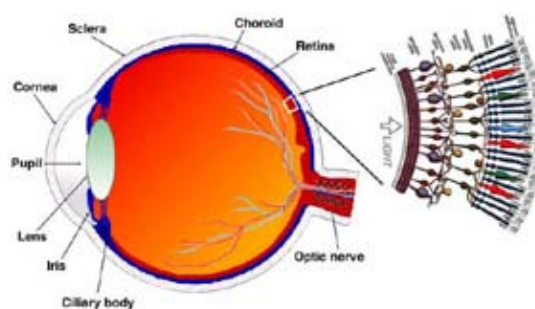
Thị giác sinh học [10]

Thị giác có lẽ là giác quan quan trọng nhất của động vật trong tự nhiên. Nghiên cứu các cơ chế của thị giác sinh học sẽ giải quyết được nhiều vấn đề về xử lý ảnh khác, có ý nghĩa ứng dụng to lớn vào trong nhiều lĩnh vực ứng dụng công nghệ.

Nguyên lý hoạt động của thị giác được mô tả như sau: Ánh sáng đi qua một lớp mô khúc xạ trong suốt gọi là giác mạc (*cornea*). Giác mạc tập trung ánh sáng và hướng nó tới con ngươi, hay gọi là lòng đen của mắt (*pupil*). Phía sau con ngươi là thấu kính, sẽ điều chỉnh sự hội tụ trên võng mạc (*retina*). Võng mạc bao gồm vài lớp tế bào.

Các tế bào kích thích nhận ánh sáng (*photoreceptors*), nằm ở lớp ngoài cùng của võng mạc, gọi là các tế bào hình que (*rods*) và tế bào hình nón (*cones*). Các tế bào hình que được tập trung ở bên ngoài, nhạy hơn đối với ánh sáng yếu. Ngược lại, các tế bào hình nón tập trung tại các hố, nhạy cảm hơn đối với ánh sáng có cường độ mạnh. Các tế bào hình que và tế bào hình nón đều chứa các thành phần hoá học biến đổi ánh sáng thành tín hiệu điện và gửi tín hiệu này tới các tế bào lưỡng cực (*bipolar*). Sau đó, các tế bào lưỡng cực chuyển qua các khớp thần kinh đến các tế bào hạch (*ganglion*), và được tập hợp lại để chuyển đến các sợi thần kinh thị giác (*optic nerve*). Quá trình xử lý sơ bộ được thực hiện ở các tế bào lưỡng cực và tế bào hạch, cho phép nhận biết chuyển động và biên của đối tượng. Bởi vì tính chất của thấu kính, ảnh của đối tượng sẽ bị lộn ngược, do đó cần phải khôi phục lại trước khi đưa đến vỏ não. Dọc theo đường dẫn đến vỏ não, ảnh từ hai mắt sẽ được trộn lẫn và tách ra để đưa đến hai bán cầu não.

Trong thị giác sinh học, phần xử lý đầu tiên là võng mạc, nó không chỉ là bộ phận thu nhận kích thích ánh sáng đơn thuần mà còn là bộ tiền xử lý đặc trưng mềm dẻo với một đầu vào liên tục và nhiều kênh thông tin đầu ra. Những kênh thông tin này liên tục lọc hình ảnh môi trường và gửi dữ liệu thời gian thực lên não. Võng mạc thực hiện một số nhiệm vụ xử lý như làm nổi biên, xử lý thích nghi mức nhiễu và ánh sáng, xử lý màu sắc, phát hiện chuyển động, chuyển



Hình 38: Cấu trúc mắt người và các thành phần của võng mạc.

đổi mã hoá tương tự thành xung... Khi thị giác bị tổn thương ở phần võng mạc, ánh sáng không được tiếp nhận và xử lý, do vậy, người ta sẽ không thể nhìn thấy được.

Các nghiên cứu gần đây đang được thực hiện bởi các nhóm nghiên cứu trong lĩnh vực cấy ghép mắt nhân tạo. Dự án nghiên cứu được tài trợ bởi công ty Optobionics Company của Mỹ, đang thử nghiệm cấy ghép chip võng mạc nhân tạo ASR (artificial silicone retina). Nhóm nghiên cứu tại đại học University of Southern California đưa ra chip MARC (Multiple-unit Artificial Retinal Chipset) vào cuối năm 2005 đủ để đọc chữ và nhận dạng mặt người. Đầu năm 2006, các nhà nghiên cứu tại Đức đưa ra công bố về kết quả thử nghiệm lâm sàng của chip võng mạc nhân tạo. Các nhà nghiên cứu của Mỹ tại các trường đại học North Carolina State University, University of North Carolina, Johns Hopkins University đã tạo ra được Chip võng mạc nhân tạo có thể cấy ghép được (Artificial Retina Component Chip - ARCC)... Nguyên lý của võng mạc nhân tạo là: ánh sáng và hình ảnh đi vào con người sẽ đi qua chip điện tử, sau đó được thu nhận và biến đổi thành các xung điện tương ứng với cường độ sáng, giống như cơ chế hoạt động của các tế bào hình que và hình nón. Xung điện này sẽ kích thích các sợi thần kinh phía sau võng mạc, và gửi thông tin lên não. Thiết bị này cho phép người khiếm thị cảm nhận được những hình ảnh thô sơ của môi trường.

Gần đây, các nhà nghiên cứu của Hungary và Mỹ ứng dụng công nghệ mạng nơ-ron tế bào CNN trong lĩnh vực mắt nhân tạo và có một số kết quả bước đầu. Công nghệ này đã mở ra một hướng nghiên cứu mới trong lĩnh vực xử lý ảnh và thị giác nhân tạo liên quan do áp dụng một số ưu điểm nổi bật của công nghệ CNN.

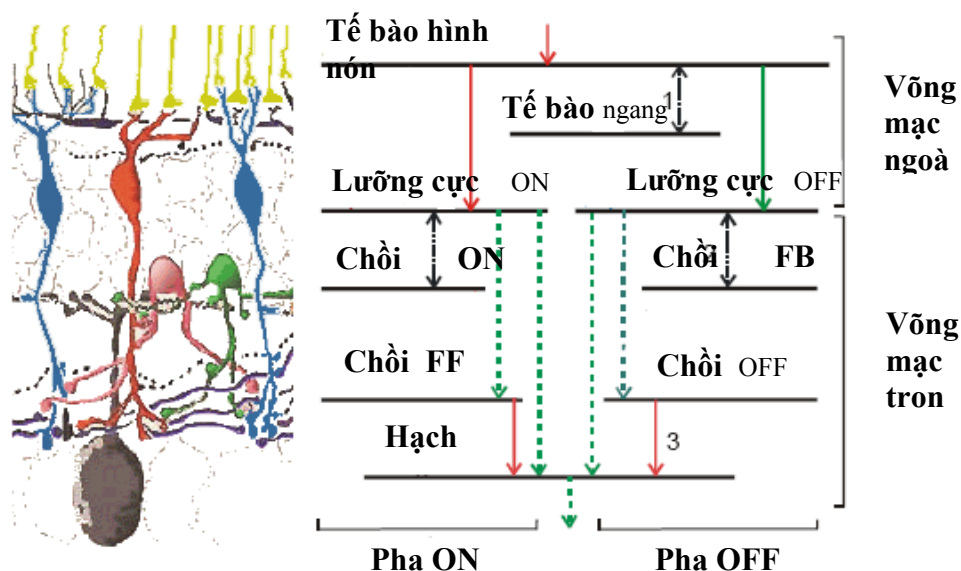
Mô hình võng mạc nhân tạo với công nghệ mạng CNN nhiều lớp[11]:

Mạng nơ-ron tế bào (Cellular Neural Networks - CNN) là một hệ xử lý song song được giới thiệu đầu tiên bởi các nhà khoa học Mỹ, từ đó đến nay có rất nhiều ứng dụng và khái niệm mới trong nhiều lĩnh vực liên quan đến CNN đã xuất hiện. Khả năng xử lý 10^{12} phép tính/giây với hàng chục ngàn CPU tích hợp song song trên một chip duy nhất, và các ứng dụng của công nghệ này cho phép giải quyết nhiều bài toán xử lý thời gian thực mà từ trước đến nay chưa làm được. Một ứng dụng tiêu biểu của CNN là máy tính xử lý ảnh nhanh tựa sinh học Bi-I được áp dụng cho các hệ thống xử lý ảnh nhanh có tốc độ xử lý trên 10 000 ảnh/giây .

Đặc điểm chính của CNN là vị trí các liên kết trong mạng: sự khác biệt chính giữa CNN và các mô hình mạng nơ-ron khác là thông tin chỉ được trao đổi trực tiếp thông qua các tế bào lân cận. Thông tin giữa các tế bào cách xa nhau được thực hiện thông qua chuỗi các tế bào liên kề.

Mô hình võng mạc nhân tạo:

Theo quan điểm kỹ thuật, võng mạc bao gồm 2 phần chính: võng mạc ngoài (*outer retina*) và võng mạc trong (*inner retina*) như hình 38. Thành phần của võng mạc ngoài là giống nhau, cung cấp dữ liệu cho tất cả các quá trình xử lý của võng mạc trong. Võng mạc trong có một số loại tế bào hạch khác nhau với các lớp xử lý riêng của mỗi loại. Hai đầu vào định tính của một tế bào hạch là thành phần ức chế (*inhibition*) và thành phần kích thích (*excitation*). Thành phần ức chế đến từ các tế bào chồi, thành phần kích thích đến từ các tế bào lưỡng cực. Mỗi tế bào chồi có ít nhất một liên kết với một tế bào lưỡng cực. Đầu vào của các tế bào lưỡng cực là từ võng mạc ngoài. Võng mạc ngoài bao gồm sự liên kết của các lớp tế bào hình nón và tế bào ngang.



Hình 39. Các kiểu liên kết noron trong võng mạc và mô hình CNN.

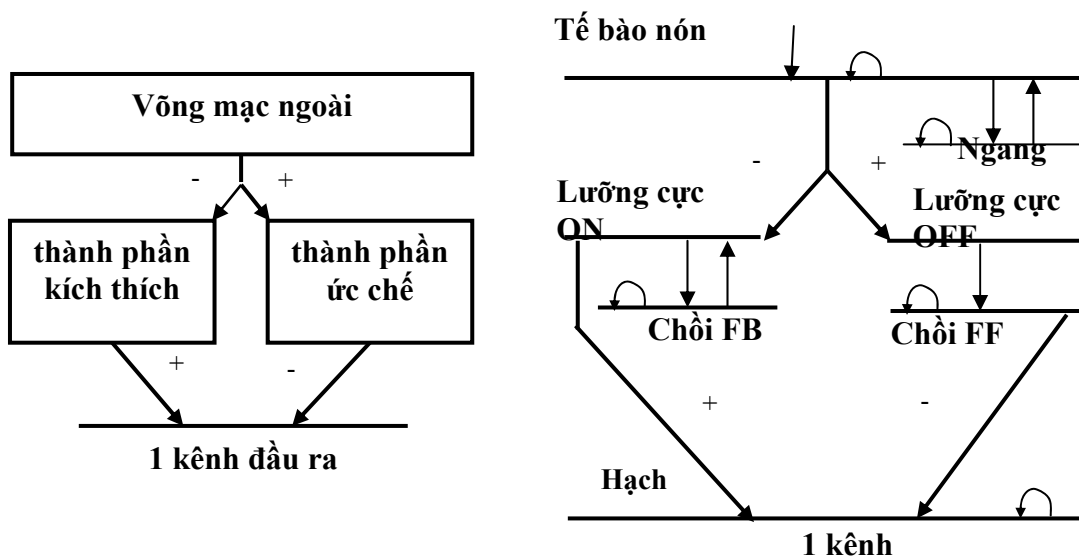
Giai đoạn xử lý thông tin đầu tiên là tại võng mạc ngoài. Giai đoạn này có thể chia thành hai pha: pha- ON và pha-OFF. Cả hai pha đều chứa một lớp tế bào lưỡng cực. Lớp này có vai trò chuyển đổi và đưa đầu ra của võng mạc ngoài tới đầu vào kích thích của lớp tế bào hạch. Đầu vào ức chế của tế bào hạch xuất phát từ một hoặc nhiều tế bào chôi.

Có hơn mười loại tế bào hạch khác nhau trong võng mạc động vật, chúng hình thành các kênh đầu ra song song để đồng thời đưa thông tin lên não. Phân loại các tế bào hạch dựa trên đặc tính điện sinh lý, bằng cách sử dụng các điểm sáng như nguồn kích thích. Nếu tế bào phản ứng trong khi chiếu sáng được gọi là tế bào On-cell, nếu nó phản ứng sau khi nguồn sáng kích thích gọi là tế bào Off-cell, và nếu nó phản ứng với cả hai trường hợp, được gọi là tế bào On-Off. Các tế bào hạch đóng vai trò như các phần tử tìm kiếm đặc trưng, quá trình xử lý được xây dựng từ các phép chuyển đổi không gian-thời gian tại các lớp lan truyền khác nhau của võng mạc. Mỗi lớp lan truyền gắn với một hằng số không gian và hằng số thời gian.

Mỗi pha song song trong võng mạc bao gồm 3 cặp liên kết giữa các lớp lan truyền. Cặp liên kết có phản hồi thứ nhất là giữa các tế bào hình nón và tế bào ngang, cặp thứ hai là liên kết có phản hồi giữa các tế bào lưỡng cực và tế bào chôi, cặp liên kết truyền thẳng thứ ba là giữa tế bào chôi và tế bào hạch. Hàm truyền giữa các lớp là hàm tuyến tính hoặc phi tuyến.

Dựa trên đặc điểm tổ chức của võng mạc ta thấy nó tương đối phù hợp với cấu trúc của mạng noron tế bào. Các nhà khoa học Hungary và Mỹ đã nghiên cứu mô hình võng mạc dựa trên cấu trúc mạng noron tế bào, và đặc điểm giải phẫu học và sinh lý học. Để xây dựng một mô hình CNN nhiều lớp, dựa trên cấu trúc tính toán của võng mạc, phải xác định các phần tử của mô hình CNN. Các phần tử này được đưa ra từ việc đo lường các thông số võng mạc của động vật. Mỗi tập hợp noron của võng mạc được mô hình hoá bởi một lớp CNN và mỗi khớp thần kinh liên kết giữa các noron được thể hiện bởi một bộ trọng số CNN xác định. Đầu vào của võng mạc là các kích thích của lớp tế bào thu nhận kích thích sáng, và đầu ra lấy từ đầu ra của các tế bào hạch.

Các khối cơ sở của mô hình võng mạc CNN gọi là các noron mô phỏng. Các noron này được tổ chức thành các lớp tách biệt. Thông số của các lớp này bao gồm các hằng số thời gian, hệ số lan truyền giữa các lớp, các hằng số không gian, các hàm đầu ra của noron.



Hình 40. Cấu trúc xử lý của một mô hình võng mạc.

Các kiểu noron khác nhau được tổ chức thành các lớp CNN 2 chiều (các đường nằm ngang trên hình 40). Một noron trong một lớp này ảnh hưởng tới các noron ở lớp khác thông qua các mô hình khớp thần kinh tuyến tính (các mũi tên), và bản thân các lớp có hằng số thời gian và không gian khác nhau (hình tròn).

Một mô hình võng mạc CNN như vậy chứa ba lớp:

- + Lớp liên kết có phản hồi giữa tế bào hình nón và tế bào ngang (cone-horizontal) mô hình hoá võng mạc ngoài.
- + Lớp liên kết có phản hồi giữa tế bào lưỡng cực và tế bào chôi (bipolar-amacrine) tạo ra thành phần kích thích.
- + Lớp liên kết truyền thẳng giữa tế bào lưỡng cực và tế bào chôi (bipolar-amacrine) tạo ra thành phần ức chế.

Phân tích mô hình võng mạc CNN:

Mô hình của võng mạc nhân tạo thể hiện bằng các phương trình vi phân phi tuyến. Các lớp chứa các phần tử phi tuyến dạng RC. Các đặc tính đầu ra là hàm sigmoid $f(\cdot)$ và điều kiện biên có thể bằng không hoặc hằng số. Mô hình này có thể nhúng trong máy tính đa năng CNN-UM.

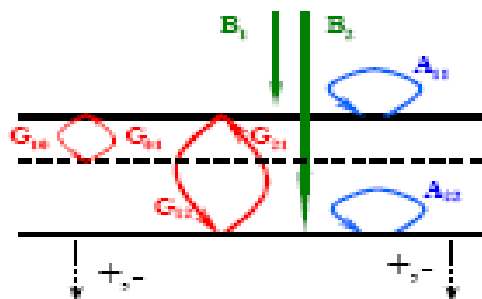
Phương trình vi phân của các lớp:

Lớp 1:

$$C_1 \dot{x}_{1,ij} = -x_{1,ij} / R_1 + g_{01} x_{10,ij} + \sum_{kl \in N_1} A_{11,kl} y_{1,kl} + g_{21} y_{2,ij} + b_1 u_{1,ij} + z_1$$

$$C_{10} \dot{x}_{10,ij} = -x_{10,ij} / R_{10} + g_{10} x_{1,ij}$$

$$y_{1,ij} = f(x_{1,ij})$$



Hình 41: Cấu trúc mô phỏng của võng mạc CNN.

Lớp 2:

$$C_2 \dot{x}_{2,ij} = -x_{2,ij} / R_2 + \sum_{kl \in N_1} A_{22,kl} y_{2,kl} + g_{12} y_{1,ij} + b_2 u_{2,ij} + z_2$$

$$y_{2,ij} = f(x_{2,ij})$$

Lớp 3:

$$y_{3,ij} = f(x_{1,ij}, x_{2,ij})$$

Trong đó:

$$\tau_1 = R_1 C_1, \quad \tau_{10} = R_{10} C_{10}, \quad \tau_2 = R_2 C_2, \quad 1 \leq i \leq M, \quad 1 \leq j \leq N,$$

Các đặc trưng được lọc ra thông qua các bộ trọng số. Liên kết giữa các tế bào trong một lớp được thể hiện bằng ma trận trọng số A, liên kết giữa các lớp G, các hệ số ngưỡng z, ma trận điều khiển B có dạng như sau:

$$A_{ij} = \begin{vmatrix} a & 2a & a \\ 2a & c & 2a \\ a & 2a & a \end{vmatrix}, \quad \text{Trong đó } \begin{cases} c = 1 - 12a, & \text{nếu } i = j \\ c > 2a, & \text{nếu } i \neq j \end{cases}$$

$$f(x) = \frac{|x+1| - |x-1|}{2}$$

$$G_{10} = g_{10}; \quad G_{01} = g_{01}; \quad G_{12} = g_{12}; \quad G_{21} = g_{21};$$

$$B_1 = b_1; \quad B_2 = b_2; \quad z_1; \quad z_2;$$

$$\tau_1; \quad \tau_{10}; \quad \tau_2$$

Dựa trên mô hình đã được đơn giản hoá này, các nhà nghiên cứu Hungary đã thử nghiệm trên chip CNN. Bằng cách thay đổi các trọng số, các hằng số thời gian và không gian để xây dựng một số phép xử lý ảnh đơn giản. Toàn bộ mô hình được kết hợp bởi một số đơn vị mẫu có thể lập trình được, gọi là Complex R-units. Mô hình này thể hiện một hệ tính toán phức tạp và hiệu quả cho một số chức năng mô phỏng quá trình xử lý hình ảnh ở võng mạc và mở ra một khả năng ứng dụng mới trong việc chế tạo “camera thị giác”, cũng như các thiết bị nhân tạo có thể cấy ghép vào các hệ sinh học khác.

Kết luận:

Võng mạc nhân tạo, có vai trò như một thiết bị tính toán là một hệ phức tạp và mềm dẻo cho quá trình tiền xử lý một chuỗi ảnh video. Chính vì vậy, nó được dự báo là có thể sử dụng trong một số thuật toán phức tạp và các ứng dụng tiếp cận đến cuộc sống, chẳng hạn như phân loại và nhận dạng, bám mục tiêu, cảnh báo... Các hướng nghiên cứu, ứng dụng mô hình võng mạc CNN trong một số lĩnh vực được xác định như sau:

- Xử lý ảnh màu RGB thời gian thực
- Phân tích ảnh động
- Phân loại và nhận dạng đối tượng
- Phân tích sự thay đổi đột ngột của điều kiện môi trường .

1.5.4. Phương pháp xử lý ảnh vân tay sử dụng mạng CNN

Mở đầu

Hiện nay, có nhiều phương pháp được ứng dụng cho nhận dạng cá nhân được quan tâm rất nhiều trong xã hội hiện đại. Tuy nhiên, nhận dạng vân tay là hướng nhận dạng sinh trắc học chuẩn và phổ biến. Vân tay có đặc trưng cá nhân và không thay đổi. Nó được sử dụng rộng rãi trong các sản phẩm bảo mật như quản lý vào ra công sở, máy rút tiền tự động ATM, hệ điều tra tội phạm, ...v.v. Vấn đề nhận dạng vân tay với cơ sở dữ liệu lớn đòi hỏi tốc độ nhanh mà vẫn đảm bảo độ chính xác cao là rất cấp thiết đối với việc quản lý con người trong một xã hội văn minh.

Trong vài năm trở lại đây, từ năm 2000 đến nay, một số cơ sở nghiên cứu khoa học trong nước như Viện KH & CN Việt Nam, trường Đại học Bách khoa Hà Nội, Học viện Kỹ thuật Quân sự, Đại học Quốc Gia Hà Nội, trường Đại học Bách khoa HCM, Bộ Công an, Bộ Bưu chính Viễn thông ... đã thực hiện nhiều đề tài nghiên cứu nhận dạng vân tay ứng dụng vào lĩnh vực bảo mật, truy tìm tội phạm, an toàn mạng thông tin, ...v.v.

Một số thuật toán phổ biến được sử dụng như “*nhận dạng ảnh vân tay đa cấp xấp xỉ với thuật toán tự động xác định điểm đặc trưng dựa vào dòng chảy đường vân*” [22] dựa trên hệ thống số phân loại Henri hoặc “*thuật toán đối sánh tuần tự các điểm đặc trưng của vân tay*” đã được nghiên cứu ở Bộ Công an.

Trung tâm Công nghệ thông tin - CDiT (Tổng công ty Bưu chính viễn thông VN) vừa công bố nghiên cứu thành công phần mềm bảo mật ứng dụng công nghệ nhận dạng vân tay vào các hệ thống tính cước, hệ thống quản lý mạng viễn thông và thiết bị trên mạng viễn thông.

Nhưng các đề tài này phát triển các thuật toán xử lý nhận dạng vân tay trên các máy vi tính xử lý tuần tự. Hiện nay, công nghệ xử lý ảnh song song CNN với tốc độ cao đang được nghiên cứu phát triển mạnh trên thế giới. Ứng dụng công nghệ mạng nơ-ron tế bào CNN vào vấn đề nhận dạng vân tay có thể nâng cao tốc độ nhận dạng đáp ứng yêu cầu với cơ sở dữ liệu vân tay lớn [39].

Một số thuật toán nhận dạng vân tay

Đặc điểm nhận dạng vân tay

Vân tay được tạo bằng đỉnh phần trên da bàn tay và chân của tất cả mọi người và một số động vật. Những đỉnh này tạo các đường với cỡ và hình dạng khác nhau. Đường nào cũng dừng và cắt được gọi là đặc trưng. Số lượng đặc trưng tạo nên đặc điểm riêng của từng vân tay có thể nhận dạng được. Để bảo đảm nhận dạng vân tay chính xác ta cần 10 đến 12 điểm đặc trưng luôn luôn không thay đổi theo thời gian.



Một số kiểu của vân tay : Vân tay không chỉ là tập hợp những đường vân mà cả các đường vân cùng nhau tạo thành hình ảnh đặc trưng. Có khoảng 7 kiểu vân tay khác nhau như sau:

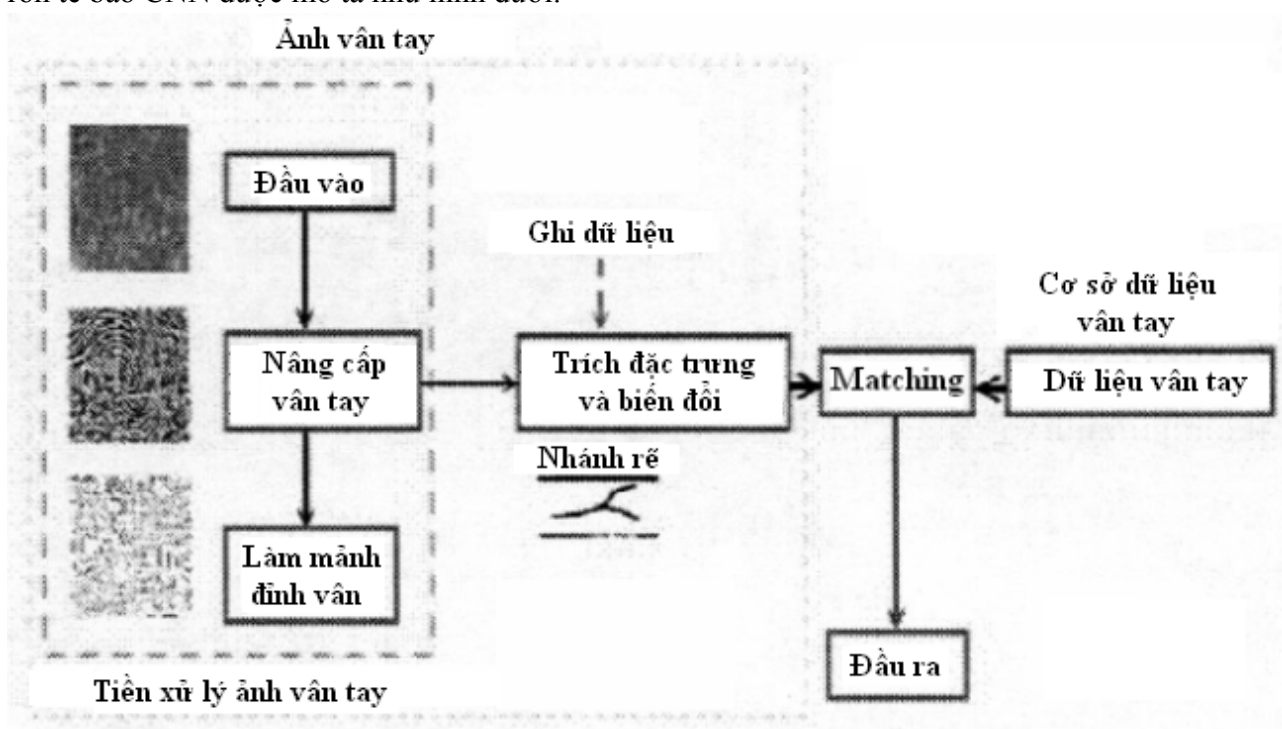
- Hình cung (Arch) : các đường chạy giống như các sóng từ một nơi đến nơi khác.

Một số kiểu của vân tay

- Tentarch : giống như Arch nhưng với que mọc ở giữa.
- Cuộn (Loop) : các đường chạy từ một nơi quay trở về giữa mặt phẳng giống nhau.
- Cuộn kép (Double loop) : giống như Loop nhưng với hai vòng mặt trong, một cái đứng một cái xuống dốc.
- Cuộn lỗ chỗ (Pocked loop) : giống như Loop nhưng với vòng tròn nhỏ ở điểm quay.
- Vòng xoắn (Whorl) : các đường tạo thành vòng tròn.
- Hình hỗn hợp (Mixed figure) : gồm có các hình khác nhau.

Thuật toán nhận dạng vân tay dùng CNN [25]

Gần đây, các bài báo đã được công bố dùng CNN để nhận dạng vân tay hầu hết đều tập trung vào vấn đề làm nổi vân tay và làm mảnh mấu. Tuy nhiên, cấu trúc đầy đủ của phương pháp nhận dạng vân tay được cung cấp ít hơn. Phương pháp nhận dạng vân tay bao gồm việc làm nổi vân tay, làm mảnh đỉnh, trích chọn đặc trưng, matching và nhận dạng sử dụng mạng nơ ron tế bào CNN được mô tả như hình dưới.



Hệ thống nhận dạng vân tay.

Nâng cấp ảnh vân tay (*Fingerprint Enhancement*) và làm mảnh đỉnh (*Ridgeline Thinning*) được thực hiện trước khi matching. Sau đó, trích chọn đặc trưng (*feature extraction*), ghi nhánh rẽ và biến đổi (*transformation*) được hoàn thành. Cuối cùng kết quả được so sánh với cơ sở dữ liệu nhận dạng vân tay.

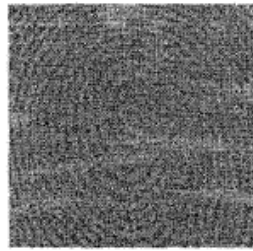
Nâng cấp ảnh vân tay đa mức xám sử dụng CNN [13]

Nâng cấp ảnh vân tay đa mức xám bao gồm co giãn độ tương phản, lọc CNN Gabor-Type. Một histogram ảnh vân tay là một đồ thị biểu hiện sự phân bố tương ứng của cường độ mức xám trong một ảnh. Nếu f_{\min} và f_{\max} là cường độ mức xám thấp nhất và cao nhất của hình ảnh ($f_{\min}, f_{\max} \in [-1, 1]$), thì những ảnh xạ sau sẽ đưa ảnh nằm trong khoảng $[f_{\min}, f_{\max}]$ tương ứng với $[-1, 1]$:

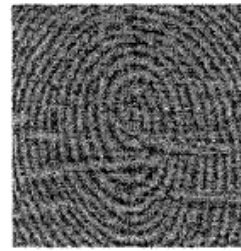
$$g(f) = -1 + 2 \frac{f - f_{\min}}{f_{\max} - f_{\min}} = \frac{2}{f_{\max} - f_{\min}} f + \frac{f_{\min} + f_{\max}}{f_{\min} - f_{\max}}$$

Mẫu CNN tương ứng là:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{2}{f_{\max} - f_{\min}} & 0 \\ 0 & 0 & 0 \end{bmatrix}; I = \frac{f_{\min} + f_{\max}}{f_{\min} - f_{\max}}$$



Ảnh gốc



Ảnh sau khi nâng cấp Histogram CNN

Nâng cấp ảnh vân tay dùng cân bằng Histogram

Làm mảnh đỉnh vân

Để tìm đường rẽ nhánh của vân tay nên đỉnh vân phải được làm mảnh trước khi trích chọn điểm đặc trưng. Làm mảnh đỉnh vân là đưa ra ảnh vân với đường có độ rộng là 1 điểm ảnh. Ảnh làm mảnh sẽ được đưa ra khi liên tiếp tìm các điểm pixel là các điểm đường viền và xóa chúng đi. Phương pháp làm mảnh đỉnh song song bằng mạng CNN gọi là CNNThinning.

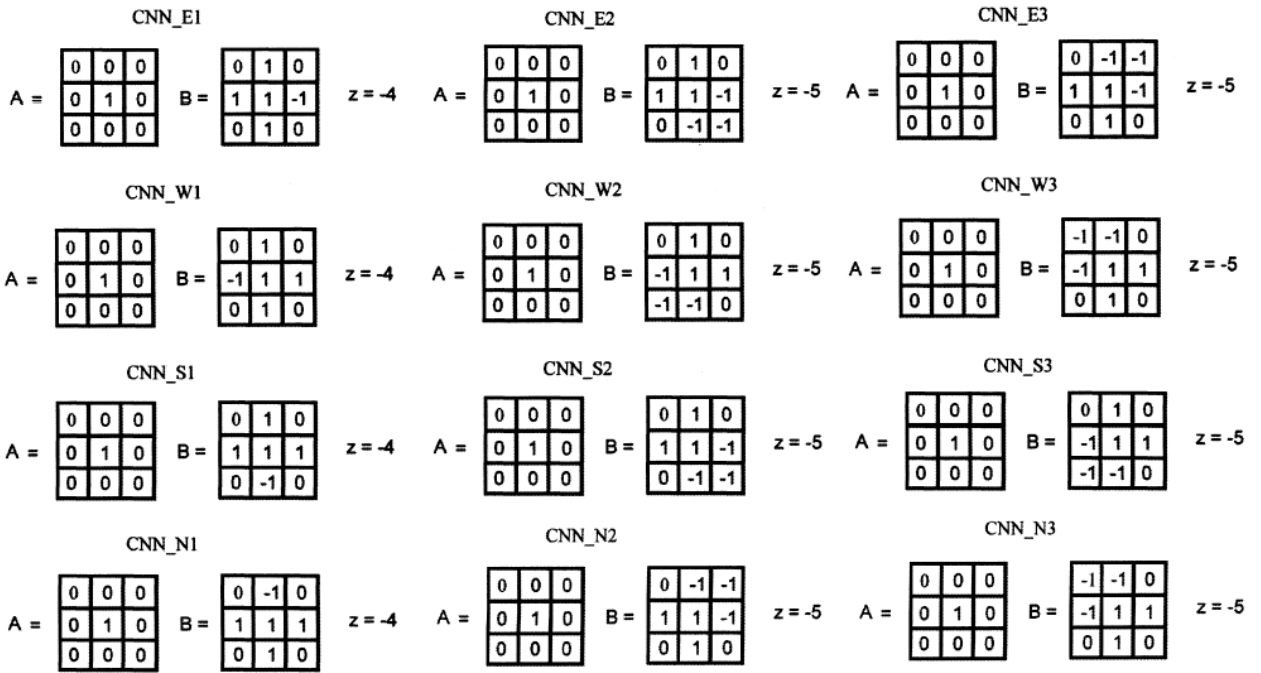
P ₁	P ₂	P ₃
P ₈	P ₀	P ₄
P ₇	P ₆	P ₅

Điểm ảnh đường viền là điểm ảnh cần xóa qua xử lý làm mảnh. Có 4 loại điểm đường viền đông, tây, nam, bắc tương ứng với từng điểm. Ví dụ như ở hình bên thể hiện điểm P₀ là điểm ảnh trung tâm,

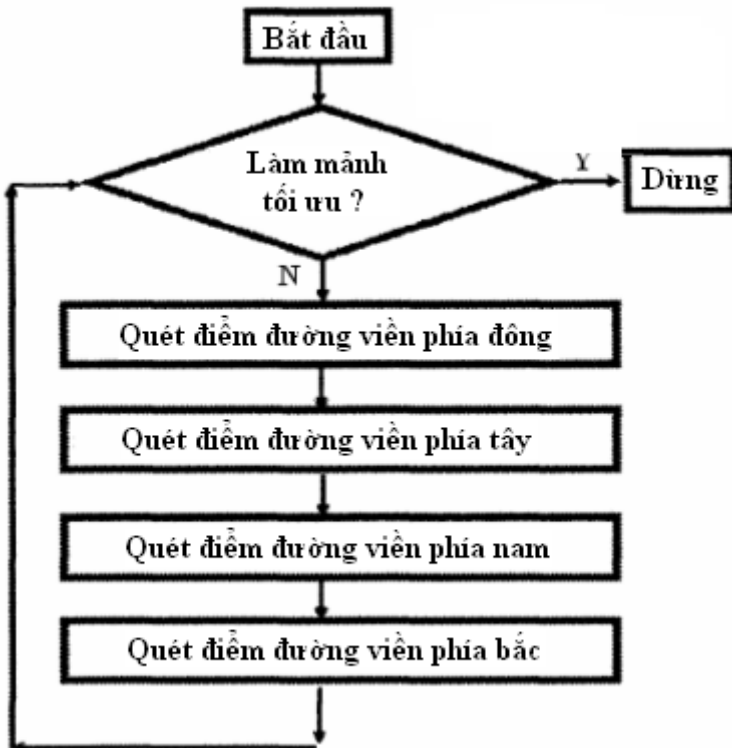
- P₄ là điểm đường viền phía đông
- P₈ là điểm đường viền phía tây
- P₆ là điểm đường viền phía nam
- P₂ là điểm đường viền phía bắc

Điểm đường viền

Ảnh được chia thành nhiều block và kích thước của block là 3x3. Một hướng của điểm đường viền có 3 kiểu template (ví dụ điểm đường viền phía đông có 3 template CNN_E1, CNN_E2, CNN_E3). Nếu block được match với mẫu bất kỳ, điểm pixel sẽ bị xóa. Ảnh làm mảnh sẽ được đưa ra nếu liên tiếp tìm các điểm pixel là các điểm đường viền và xóa chúng đi.



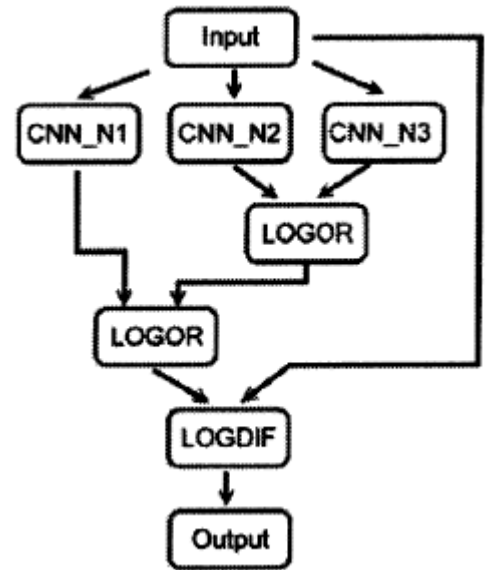
Các mẫu của các điểm đường viền có tham số CNN (A, B, z)



Sơ đồ khối làm mảnh đỉnh vân

Trích chọn đặc trưng vân tay

Có hai đặc trưng cơ bản được trích từ ảnh vân tay là *đỉnh kết thúc của đường vân* và *nhánh rẽ vân tay*. Phương pháp xác định vùng đặc trưng ảnh vân tay được A.Wahab, S.H.Chin, E.C.Tan công bố năm 1998. Trong đó, xác định cửa sổ 3x3 (xem hình bên), M là điểm dò và X_1, \dots, X_8 là các điểm lân cận của nó trong theo chiều kim đồng hồ bắt đầu từ góc trên-bên trái. Nếu X_n là



Sơ đồ khối của tìm kiếm điểm đường viền.

X_1	X_2	X_3
X_8	M	X_4
X_7	X_6	X_5

Cửa sổ 3x3 để trích đặc trưng

điểm ảnh màu đen thì giá trị $R(n)$ của nó sẽ là 1. Cũng như thế, $R(n)$ sẽ là 0 nếu điểm ảnh là màu trắng. Nếu M là đỉnh kết thúc, nó cần phải có:

$$C_n = \sum_{k=1}^8 |R(k+1) - R(k)| = 2 \text{ với } R(9) = 1.$$

Đối với M là nhánh rẽ, nó sẽ là :

$$C_n = \sum_{k=1}^8 |R(k+1) - R(k)| = 6$$

Trên cơ sở phương pháp trên, Te-Je Su, Yan-Yi Du, Ying-Jen Cheng and Yi-Hui Su đã đưa ra phương pháp trích đặc trưng vân tay dùng CNN. Phương pháp trích chọn đặc trưng có thể được mô tả bằng hàm sau:

$$\sum_{k=1}^8 |P(k+1) - P(k)| = 12|$$

Và các tham số CNN (A,B,z) để trích đặc trưng vân tay như sau:

CNN_Feature_Extract

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} b_1 & b_2 & b_3 \\ b_8 & U_{ij} & b_4 \\ b_7 & b_6 & b_5 \end{bmatrix} \quad z = -12$$

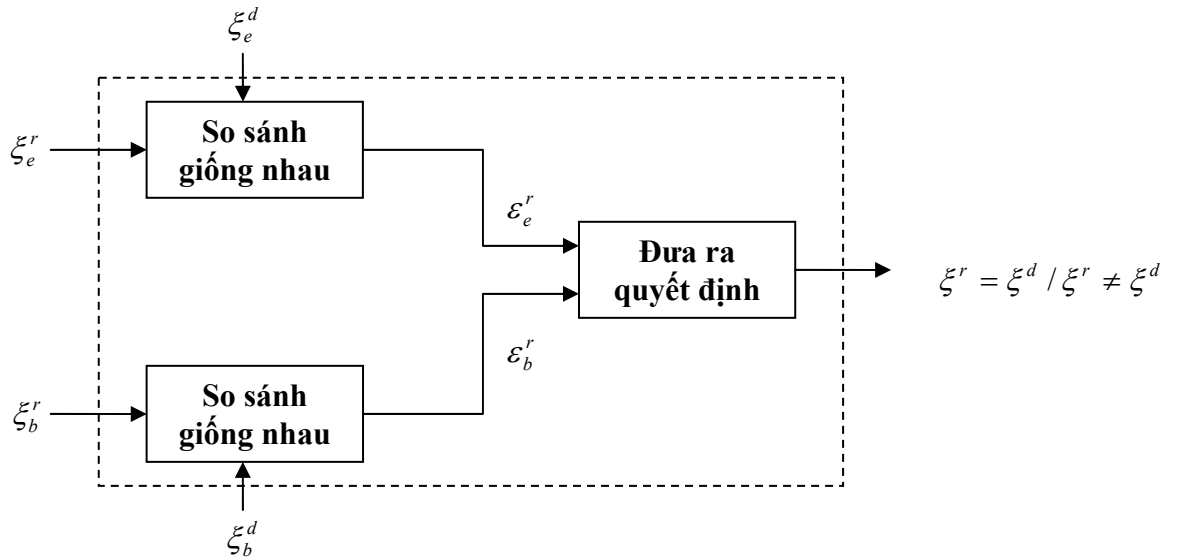
$$b_i = |U_k - U_i| \quad \text{Trong đó } k = (i+1) \bmod 8 \quad \text{với } i = 1 \sim 8$$

Tham số CNN (A,B,z) để trích chọn đặc trưng

Tất cả các điểm đặc trưng dò tìm trong ảnh được làm mảnh bao gồm tọa độ và kiểu của điểm đặc trưng được ghi vào cơ sở dữ liệu đặc trưng. Tại điểm kết thúc của việc trích đặc trưng, một bản ghi đặc trưng của vân tay được thiết lập.

Matching đặc trưng vân tay dùng CNN

Mục đích của việc matching đặc trưng vân tay là quyết định đặc trưng vân tay cần nhận dạng (ξ^r) giống với đặc trưng mẫu trong cơ sở dữ liệu ξ^d . Quyết định $\xi^r = \xi^d$ khi khoảng cách giữa ξ^r và ξ^d nhỏ hơn hoặc bằng một giá trị ngưỡng cho trước ϵ . Thuật toán Matching đặc trưng vân tay dùng CNN gồm 2 quá trình xử lý chính : So sánh đồng dạng (Similarity Comparison) và đưa ra quyết định (decision making)



Hình 42 : Sơ đồ khối quá trình matching vân tay dùng CNN

Trong đó,

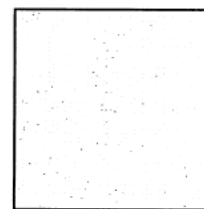
- ξ_e^r : đặc trưng điểm kết thúc của đường vân cần nhận dạng.
- ξ_b^r : đặc trưng nhánh rẽ của đường vân cần nhận dạng.
- ξ_e^d : đặc trưng điểm kết thúc của đường vân mẫu trong cơ sở dữ liệu.
- ξ_b^d : đặc trưng nhánh rẽ của đường vân mẫu trong cơ sở dữ liệu.



Làm nổi vân tay



Làm mảnh đỉnh



Trích đặc trưng

Kết quả thử nghiệm của một số phương pháp xử lý ảnh vân tay bằng CNN

Vấn đề nhận dạng vân tay với cơ sở dữ liệu lớn đòi hỏi tốc độ nhanh mà vẫn đảm bảo độ chính xác cao là rất cấp thiết đối với việc quản lý con người trong một xã hội văn minh. Đã có nhiều đề tài liên quan đến lĩnh vực này đã được thực hiện ở Việt Nam nhưng việc nghiên cứu phát triển các giải thuật xử lý song song và ứng dụng hệ thống xử lý ảnh nhanh trên nền mạng CNN cho nhận dạng nhanh vân tay thì còn chưa được quan tâm. Vì vậy, chúng tôi tiếp cận vấn đề này với mục đích tìm hiểu và phát triển phương pháp nhận dạng vân tay với một số thuật toán xử lý song song trên cơ sở mạng nơ-ron tế bào CNN ứng dụng trên hệ thống Bi-I như một cách tiếp cận mới để đáp ứng được yêu cầu nhận dạng nhanh.

1.6. Khả năng ứng dụng của CNN

1.6.1. Khả năng ứng dụng công nghệ CNN trong công nghiệp và các ngành kinh tế

Một số thách thức trong xử lý ảnh

Quan sát là một trong những phương thức thu thập thông tin quan trọng. 79% thông tin con người thu thập được qua thị giác đều bắt nguồn từ ảnh. Các hình ảnh được truyền tới bộ não, được xử lý tại đây, rồi sau đó các chỉ thị của não sẽ gây ra các phản ứng tương ứng của cơ thể. Nhưng những năng lực tự nhiên của con người không phải là vô hạn. Trong rất nhiều trường hợp khả năng bắt ảnh của mắt người, khả năng xử lý của não người không đáp ứng được các yêu cầu quan sát - Đặc biệt là trong lĩnh vực sản xuất công nghiệp, nơi mà tốc độ của các dây chuyền sản xuất cực kỳ nhanh, thời gian tồn tại sự kiện vô cùng nhỏ, số lượng đối tượng quan sát, thông tin cần xử lý vô cùng lớn. Con người với sức sáng tạo không ngừng đã tạo ra được nhiều công cụ có khả năng hỗ trợ, khắc phục được các năng lực hạn chế của bản thân. Các camera độ nhạy cao cùng với các máy tính số mạnh đã tạo ra các hệ thống quan sát có khả năng thay thế con người trong nhiều trường hợp. Tuy nhiên vẫn còn quá nhiều khoảng trống trong bức tranh xử lý ảnh bằng máy tính trong công nghiệp. Do sử dụng các máy tính số hoạt động theo nguyên lý xử lý lệnh tuần tự làm công cụ thực hiện nên mặc dù các chuyên gia xử lý ảnh đã cố gắng cải tiến, hoàn thiện các thuật toán thu và xử lý ảnh nhưng với nhiều nhu cầu như thu và xử lý ảnh trong những thời gian cỡ micro giây, tốc độ bắt ảnh và xử lý vài trăm hình trên giây (fps), xử lý ảnh thời gian thực từ nhiều nguồn ảnh tới cùng một lúc... thì phương pháp xử lý ảnh truyền thống đã không đáp ứng được. Việc tăng tốc độ xử lý của các bộ vi xử lý cũng không phải là vô hạn do các hạn chế về cấu trúc vật lý của vật liệu bán dẫn.

Mạng nơron tế bào CNN (Cellular Nonlinear/Neural Network) được các nhà bác học Mỹ Leon O.Chua và L.Yang giới thiệu năm 1988 và sau đó năm 1992 là máy tính vạn năng tương tự logic CNN UM đã mở ra một lối thoát cho bài toán xử lý ảnh thời gian thực. Với kiến trúc mạng nơron tế bào, xử lý song song với hàng chục nghìn CPU (sản phẩm thương mại hiện nay là 16.384 CPU), lập trình được từ các máy tính số sử dụng hệ điều hành thông dụng như Windows 2000, XP khả năng bắt ảnh và xử lý của các thiết bị ứng dụng công nghệ CNN đã đạt tới 10.000 fps; trong nhiều trường hợp đã đạt đến 50.000fps.

Nhu cầu xử lý ảnh trong công nghiệp

Ngày nay sản xuất công nghiệp đã đạt tới trình độ tự động hoá cao, tốc độ sản xuất của dây chuyền rất lớn. Thực tế này dẫn tới là nếu có sai lệch trong quá trình sản xuất thì chỉ trong những khoảng thời gian rất ngắn đã có ngay hàng loạt lớn số lượng sản phẩm phế phẩm. Do đó mọi khâu từ kiểm tra giám sát đến điều chỉnh đều phải đáp ứng rất nhanh; đặc biệt là khâu kiểm tra phát hiện. Nhu cầu xử lý ảnh trong công nghiệp hiện nay càng ngày càng lớn chủ yếu gồm: đo, kiểm tra, phân loại, hiển thị. Các hệ thống xử lý ảnh công nghiệp này ngoài yêu cầu chính xác còn cần phải đảm bảo yêu cầu về thời gian thực (real-time).

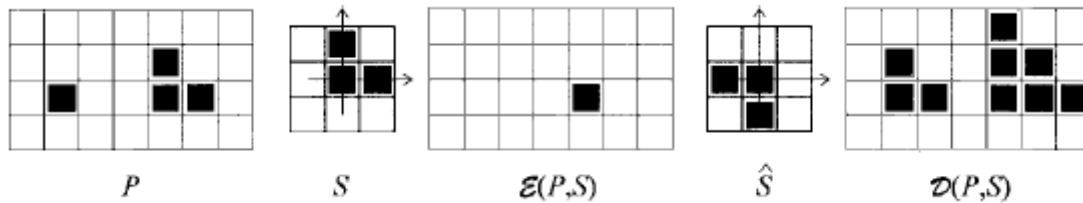
Xử lý ảnh bằng máy tính hệ lệnh tuần tự

Xử lý ảnh bằng máy tính hệ lệnh tuần tự đã đạt được nhiều thành tựu trong các lĩnh vực của thực tiễn như các chương trình nhận dạng mặt người, nhận dạng vân tay trong điều tra hình sự, các chương trình xử lý ảnh vệ tinh, kiểm soát giao thông, xử lý ảnh chụp cắt lớp, chuẩn đoán

tế bào trong y học, các chương trình nhận dạng chữ viết.vv..đã đem lại nhiều ứng dụng tiện ích cho con người. Việc nghiên cứu các phương pháp xử lý ảnh, cải tiến các thuật toán xử lý trên máy tính hệ lệnh tuần tự vẫn được tiếp tục nhằm đáp ứng các nhu cầu của thực tế. Tuy nhiên việc xử lý đã gặp phải các ngưỡng, các giới hạn khó có thể vượt qua như đã nói ở phần đầu bài viết. Vì sao vậy? Ta hãy xem xét quá trình xử lý ảnh bằng phương pháp truyền thống:

Các ảnh đưa vào máy tính để xử lý là các ảnh đã được số hoá. Chúng có thể là ảnh nhị phân (chỉ có 2 mức đen/trắng) hoặc ảnh mức xám (đen trắng có nhiều mức) hoặc ảnh màu (tổng hợp từ 3 màu cơ bản: đỏ, xanh, lục). Các ảnh được lưu trữ dưới dạng các ma trận có kích cỡ khác nhau. Khi xử lý ảnh tùy theo yêu cầu các giá trị ảnh sẽ bị biến đổi với các phép tính toán khác nhau. Việc thao tác được thực hiện lần lượt từng bit một. Ảnh càng lớn, thao tác càng phức tạp số lượng phép tính thực hiện yêu cầu càng lớn dẫn đến thời gian xử lý càng dài.

Một ví dụ thực hiện phép biến đổi hình thái học erosion & dilation ảnh nhị phân như sau:



Hình 43: Thực hiện erosion và dilation ảnh nhị phân

Ta có một ảnh P như hình 43. Thực hiện phép erosion và dilation ảnh bằng cách tùy theo yêu cầu xử lý tạo ra các tập phần tử có cấu trúc (mặt nạ), sau đó rê mặt nạ đi khắp ảnh và tính giá trị điểm ảnh bởi các điểm lân cận với motif của mặt nạ. Các phép thực hiện có thể là hội, tuyển hoặc liên hợp. Chẳng hạn Dilation: Rê mặt nạ S đi khắp ảnh và tại mỗi điểm kiểm tra nếu bit có giá trị 1 thì thực hiện phép tuyển với bit mặt nạ quanh điểm ảnh đó. Kết quả được D(P,S) Theo cách này khối lượng tính toán rất lớn: Với ảnh P như trên có 28 bit cần kiểm tra 28 lần, tại mỗi vị trí cần thực hiện 9 lần phép tuyển. Với các thao tác lọc, dò biên, trích chọn đặc tính... số lượng tính toán cũng vô cùng lớn.

Công nghệ CNN và hệ thống thu ảnh - xử lý song song

Công nghệ CNN được xây dựng trên tư tưởng chung là sử dụng một mảng đơn giản các tế bào, liên kết cục bộ để xây dựng một hệ thống xử lý tín hiệu analog lớn. Đặc điểm mấu chốt của mạng nơron là xử lý song song không đồng bộ, thời gian liên tục.

Khi mới được phát minh các CNN chỉ có thể thực hiện các nhiệm vụ cụ thể tùy theo các trọng liên kết (mẫu) của nó. Để việc sử dụng linh hoạt, mềm dẻo và linh hoạt hơn người ta đã nghĩ tới chuyện thay đổi các trọng liên kết. Kiến trúc máy tính vạn năng tương tự logic CNN UM (CNN Universal Machine) ra đời (1992). Từ đây các mẫu có thể thay đổi tùy theo yêu cầu của phép xử lý. Việc đưa bộ mẫu vào CNN UM được thực hiện thông qua máy tính số thông thường.

Tiến xa hơn nữa, để tạo ra các thiết bị chuyên dùng xử lý ảnh có công suất mạnh người ta đã tích hợp các mảng sensor thu ảnh vào thiết bị CNN. Luồng tín hiệu ảnh từ các sensor được đưa thẳng tới đầu vào của chip CNN. Quá trình động học diễn ra với những thời gian vô cùng ngắn (5 μ s hoặc nhỏ hơn). Tùy theo tập mẫu nạp vào theo những yêu cầu khác nhau của xử lý ảnh mà tín hiệu đầu ra của các cell - tập hợp thành ảnh đã được xử lý khác nhau.

Để thuận tiện cho việc sử dụng, các bộ mẫu tốt, ổn định đã được kiểm nghiệm sử dụng cho thiết bị CNN được nhà sản xuất cung cấp cho người sử dụng thông qua thư viện mẫu chẳng hạn. Việc nghiên cứu các tập mẫu (template) trong CNN là một vấn đề mấu chốt nhằm mục đích tìm ra các phương pháp tạo các gien tùy theo yêu cầu của tác vụ xử lý.

Khả năng ứng dụng công nghệ xử lý ảnh nhanh CNN trong các ngành công nghiệp [35]

Ngoài rất nhiều ứng dụng đa dạng và có ý nghĩa lớn trong các lĩnh vực an ninh quốc phòng, trong phần này chúng tôi chỉ đề cập tới các khả năng đáp ứng của công nghệ CNN với nhu cầu thực tiễn công nghiệp và một số lĩnh vực khác như là một giải pháp duy nhất.

- Ứng dụng trong công nghiệp dược: Kiểm tra, phân loại thuốc, sản phẩm y học. Tại Phòng thí nghiệm nghiên cứu CNN thuộc Viện công nghệ thông tin, thí nghiệm phân loại các viên thuốc trong sản xuất đã được thực hiện với tốc độ > 10.000fps có thể phát hiện các viên thuốc bị lỗi trên dây chuyền chuyển động rất nhanh mắt thường không biệt được.
- Ứng dụng trong chuẩn đoán y học: Xây dựng hệ thống điện tâm đồ 2D thời gian thực, điện tâm đồ 3D trực tuyến. Phân tích thời gian thực chuỗi DNA. Ứng dụng nghiên cứu miễn dịch. Chế tạo mắt nhân tạo-dự kiến năm 2015 sẽ làm ra mắt nhân tạo sử dụng công nghệ CNN.
- Trong ngành in [17]: Các sản phẩm in; nhất là các sản phẩm in cao cấp, quan trọng như tiền giấy, thẻ tín dụng, mẫu chứng chỉ, bằng cấp, nhãn mác...cần phải được kiểm tra rất kỹ để tránh các sai sót, nhất là các chi tiết chống làm giả. Hoạ tiết của sản phẩm thường phức tạp, sản xuất trên dây chuyền chuyển động nhanh (tới 4m/s). Trong những trường hợp này có thể sử dụng công nghệ CNN.
- Kiểm tra bề mặt (không tiếp xúc) sản phẩm trong khi sản xuất (Surface Inspection) là phương pháp kiểm tra mà dụng cụ đo không tiếp xúc cơ khí lên đối tượng cần đo: Các vết đen, các chỗ rách của giấy, các chỗ lỗi của các tấm nhôm, sắt, nếu có sẽ được phát hiện ngay trong quá trình sản xuất. Một ứng dụng được chúng tôi đề xuất có thể kể đến là kiểm tra kích thước hình học và bề mặt thép cán (cả thép cây và thép cuộn, thép trơn và thép hình): Việc phát hiện trực tuyến (on-line) các sai lệch kích thước hình học và độ bóng bề mặt trong sản xuất thép cán rất quan trọng. Tuy nhiên do các phôi thép trong quá trình cán rất nóng (950⁰C đến 1000⁰C) và chuyển động rất nhanh (Khoảng 20 m/s - tương đương 72 km/giờ) nên không đo trực tiếp được và các camera thường cũng không bắt được ảnh. Sử dụng công nghệ CNN kết hợp với phần mềm giám sát trong trường hợp này cho đến nay là giải pháp khả quan.
- Ứng dụng kiểm tra độ cách điện của sứ cách điện cao áp: Để kiểm tra các quả sứ cao áp, kiểm tra bugi, kiểm tra các thiết bị đóng cắt điện cần phải kiểm tra sự xuất hiện tia lửa điện trên đối tượng cần kiểm tra. Do các tia lửa điện này có thời gian tồn tại rất ngắn nên các thiết bị bắt ảnh thông thường không đáp ứng được yêu cầu mà phải sử dụng các thiết bị CNN. Tại Phòng thí nghiệm nghiên cứu CNN thuộc Viện Công nghệ thông tin - Viện khoa học Việt Nam thí nghiệm phát hiện ánh sáng có thời gian tồn tại ngắn đạt tốc độ lớn hơn 50.000fps đã được thực hiện thành công
- Ứng dụng trong cơ khí chế tạo máy: Kiểm tra, phân loại các sản phẩm cơ khí có kích thước nhỏ, số lượng lớn như đai ốc, đinh ốc, các chi tiết máy nhỏ, v.v...
- Trong vận tải hàng không: Có thể đưa ra ví dụ dùng công nghệ CNN tìm các mảnh vụn kim loại trong dầu bôi trơn cho các động cơ máy bay, nhất là các máy bay vận tải cỡ lớn. Trong quá trình làm việc, do ma sát từ các chi tiết truyền động cơ khí có thể bong ra các mảnh vỡ kim loại nhỏ. Cần phân biệt các mảnh vỡ này với bọt của dầu chuyển động bôi

tron, xác định số lượng chúng. Từ đó cho ra quyết định cảnh báo cho hệ thống có nên thay dầu hay không (*mặc dù chưa đến kỳ hạn thay*)

- Ứng dụng trong ngành sản xuất ô tô: Tạo các sensor điều khiển các túi khí thông minh bảo vệ người lái và hành khách khi xảy ra va chạm. Ứng dụng xây dựng các hệ thống phản ứng tự động trong các trường hợp khẩn cấp do tốc độ xử lý của thiết bị sử dụng công nghệ CNN rất cao.
- Trong viễn thông: Tối ưu hoá hệ thống truyền dữ liệu băng thông rộng.
- Ứng dụng trong nông nghiệp: phân loại hạt ngũ cốc, hoa quả, ... với tốc độ cao
- Cuối cùng, ngoài khả năng được dùng cho các ứng dụng có yêu cầu tốc độ và khả năng tính toán cực mạnh (như một số một số ví dụ như trên) thì các thiết bị xử lý ảnh CNN vẫn thực hiện được tất cả các ứng dụng mà các hệ xử lý ảnh bằng máy tính hệ lệnh tuần tự đã thực hiện từ trước đến nay.

1.6.2. Nhu cầu và tiềm năng ứng dụng công nghệ CNN cho quốc phòng và an ninh

Việc ứng dụng công nghệ mạng nơ-ron tế bào cho mục đích quốc phòng – an ninh cũng như cho các mục đích khác sẽ phát huy hiệu quả nếu việc ứng dụng đó thực sự cần tốc độ xử lý ảnh rất cao hoặc tốc độ tính toán rất cao. Trong trường hợp cần tốc độ tính toán cao để giải các bài toán lớn thì bài toán phải thuộc loại có thể đưa về dạng để giải bằng các thuật toán song song.

Dưới đây ta sẽ lần lượt xem xét nhu cầu và khả năng ứng dụng công nghệ mạng nơ-ron tế bào để:

- Xử lý ảnh tốc độ cao
- Xử lý thông tin song song tốc độ cao

Ứng dụng công nghệ mạng nơ-ron tế bào để xử lý ảnh tốc độ cao

a.) Khả năng ứng dụng trong các hệ thống quan sát và điều khiển hỏa lực

Hệ thống quan sát và điều khiển hỏa lực có chức năng phát hiện mục tiêu trên không và/hoặc trên biển, xác định tọa độ mục tiêu để chỉ thị mục tiêu cho các trận địa tên lửa/súng pháo phòng không hoặc tên lửa đối hải. Hệ thống này có thể làm việc trên dải sóng siêu cao tần (radar sóng cm hoặc mm), dải sóng quang học hồng ngoại ($>0,7\mu\text{m}$) hoặc khả kiến ($0,4-0,7\mu\text{m}$). Phần lớn các thiết bị quang-điện tử dùng cho mục đích này trong quân sự đều làm việc ở dải bức xạ hồng ngoại (bức xạ nhiệt) vì có thể đảm bảo làm việc được cả ban đêm không có ánh sáng mờ và có khả năng chống nhiễu tốt. Trong trường hợp làm việc ở dải quang học, để đo cự li đến mục tiêu người ta thường dùng thiết bị đo xa laser (ta không đề cập đến thiết bị này). Ở đây ta chỉ xem xét khả năng ứng dụng công nghệ xử lý ảnh tốc độ cao trong hệ thống quang-điện tử để chế tạo ra các hệ thống thiết bị quan sát và điều khiển hỏa lực có tính năng cao.

Yêu cầu đối với hệ thống trong trường hợp sử dụng tên lửa (có điều khiển) không đòi hỏi độ chính xác quá cao (có thể chấp nhận sai số đến 1-2 độ), tuy nhiên yêu cầu đối với hệ thống trong trường hợp sử dụng súng pháo (đạn không điều khiển) lại đòi hỏi độ chính xác rất cao (chỉ có thể chấp nhận sai số khoảng vài phần trăm độ). Yêu cầu về góc quan sát (đối với hệ thống trong tất cả các trường hợp) càng rộng càng tốt, thông thường ít nhất cũng phải vài chục độ về góc hướng và góc tà. Yêu cầu về cự li phát hiện tối đa càng xa càng tốt, đối với hệ thống trong trường hợp sử dụng tên lửa vào khoảng vài km đến vài chục km trở lên, đối với hệ thống trong trường hợp sử dụng súng pháo có thể chấp nhận cự li phát hiện vài km. Phần lớn tên lửa hiện nay đều thuộc loại “phóng và quên”, tức là sau khi phóng tên lửa tự bay đến mục tiêu mà không

cần điều khiển từ xa như các tên lửa thế hệ cũ (SAM-2, SAM-3 ...). Vì vậy tần số cập nhật thông tin về mục tiêu không đòi hỏi quá lớn (chu kì cập nhật có thể là một vài giây).

Thiết bị sử dụng công nghệ mạng neuron tế bào có khả năng nhận và xử lí 10.000 ảnh/s đối với ảnh 128x128 pixel nhận và xử lí 28 ảnh/s đối với ảnh 1280x1024 pixel. Để có thể xem xét khả năng ứng dụng trong các hệ thống quan sát và điều khiển hoả lực ta cần quan tâm đầu tiên đến cự li tối đa D_{max} có thể phát hiện được các mục tiêu. Rõ ràng là khi mục tiêu ở rất xa, ảnh mục tiêu trên ma trận ảnh chỉ được thể hiện bằng một vài pixel thì không thể phân biệt được mục tiêu với nền. Để xác định được cự li tối đa D_{max} ta cần giải bài toán sau đây.

Tính cự li tối đa D_{max} có thể phát hiện được các mục tiêu có kích thước đặc trưng chiều ngang là L . Giả thiết hệ quang học của camera có góc mở là A [rad], độ phân giải của ảnh theo chiều ngang là N [pixel] (chiều ngang của ma trận ảnh) và mục tiêu có thể phát hiện được khi có ảnh của của mục tiêu nói trên theo chiều ngang với số pixel là $n \geq n_{min}$ [pixel]. Góc nhìn của vật có kích thước L ở cự li D là (với giả thiết $D \gg L$): $a = L/D$. Do đó kích thước L sẽ tương ứng với số pixel $n = a.N/A$ trên ma trận ảnh (đối với góc mở khá nhỏ).

Từ điều kiện $n \geq n_{min}$ ta có:

$$D \leq D_{max} = (L.N)/(A.n_{min})$$

Để có thể hình dung cụ thể trên bảng 1 có tính một vài ví dụ cho trường hợp $n_{min} = 10$ pixel, với góc mở khác nhau: $A=0,05$ rad ($\sim 3^\circ$), $A=0,1$ rad ($\sim 6^\circ$) và $A=0,15$ rad ($\sim 9^\circ$).

Bảng 1. Cự li phát hiện tối đa D_{max} [m]

A, Rad	N, pixel	L=3 m (mục tiêu cỡ xe ô tô)	L=10 m (mục tiêu cỡ máy bay)	L=100 m (mục tiêu cỡ tàu biển)
0,05	128	786	2.560	25600
0,05	1280	7.860	25.600	256.000
0,10	128	384	1.280	12.800
0,10	1280	3.840	12.800	128.000
0,15	128	256	853	8.533
0,15	1.280	2.560	8.533	85.333

Từ bảng 1 ta thấy cự li phát hiện một vật có kích thước nhất định sẽ càng xa nếu góc mở A càng nhỏ và độ phân giải của ma trận ảnh N càng lớn. Trong thực tế cự li D_{max} còn phụ thuộc nhiều vào điều kiện thời tiết, độ cong của trái đất, độ chiếu sáng, dải phổ quang học (dải khả kiến, dải hồng ngoại, ...). Tuy nhiên, những tính toán trên cho ta thấy định hướng ứng dụng: nên sử dụng cho các phương tiện phòng không (mục tiêu là máy bay, tên lửa ...) và phương tiện đối hải (mục tiêu là tàu biển) vừa là những mục tiêu quan trọng có giá trị lớn, vừa là những mục tiêu có kích thước khá lớn nên có thể phát hiện từ cự li vài km đến vài chục km. Ngoài ra góc mở của hệ quang học (trường nhìn) chỉ nên sử dụng các góc nhỏ (khoảng dưới 10°).

Vì vậy để đồng thời mở rộng trường nhìn mà vẫn đảm bảo cự ly phát hiện xa ta phải quay camera có trường nhìn hẹp quét theo một chương trình nhất định. Tuy nhiên tốc độ quét lại phụ thuộc vào tốc độ xử lý ảnh. Nếu xử lí ảnh trên máy tính theo công nghệ truyền thống (khoảng vài chục ảnh/s) thì tốc độ quét không thể quá một vài độ/giây. Và chu kì chuyển động quét (tần số cập nhật thông tin) có thể lên đến vài chục giây, rõ ràng không thể chấp nhận được đối với trạm quan sát và điều khiển hoả lực phòng không.

Như vậy, ví dụ có một thiết bị quang-điện tử có góc nhìn không gian là 6° với tốc độ xử lí 20 ảnh/s, hệ số trùng khớp không ít hơn 0,9 (ảnh trước lệch với ảnh liền ngay sau không quá $0,6^\circ$) thì tốc độ quét không thể quá $0,6^\circ/\text{ảnh} \times 20 \text{ ảnh/s} = 12^\circ/\text{s}$. Nếu phải quan sát một không gian có góc tà 24° và góc phương vị 60° với các tham số nói trên, giả thiết là hệ số trùng khớp theo góc tà không ít hơn 0,6 tức là cần 10 dòng quét theo từng góc tà lệch nhau không quá

$2,4^0$, ta sẽ thấy cần phải có thời gian là $10 \times 60 \text{ độ} / 12 \text{ độ/s} = 50\text{s}$ tức là gần 1 phút mới xong được một chu kì quan sát. Điều đó có nghĩa là cứ sau 50s thông tin về vùng quan sát mới được cập nhật một lần. Chu kì quan sát đó rõ ràng là không thể chấp nhận được đối với trạm quan sát phòng không.

Với công nghệ mạng nơron tế bào, tốc độ xử lí ảnh có thể đạt 10.000 ảnh/giây tức là gấp khoảng 500 lần so với tốc độ xử lí ảnh trên máy tính thông thường. Tương ứng với các tham số trong ví dụ trên, chu kì quan sát có thể giảm xuống còn khoảng 0,1s (thông tin được cập nhật rất nhanh, có thể coi là hệ thống bám sát đa mục tiêu dùng để điều khiển hỏa lực-fire control multitarget tracking system). Tùy theo yêu cầu về chu kì và góc quan sát, ví dụ là không quá 1s ta có thể mở rộng trường nhìn đến 150^0 (2,5 lần) mà vẫn sử dụng gúc nhỡn nhỏ để đảm bảo cự li phát hiện mục tiêu.

Trên đây là ví dụ minh họa, để thiết kế chế tạo thiết bị thật thì cần cân nhắc kĩ các nhiệm vụ thiết kế và cần lựa chọn độ phân giải phù hợp (để có tốc độ xử lí ảnh tương ứng).

Với công nghệ mạng nơron tế bào và ứng dụng phương pháp đề xuất nêu trên ta có thể đạt các chỉ tiêu khá cao (tương đương thiết bị trong phụ lục 2), kích thước trọng lượng gọn nhẹ (không quá vài chục kg) và giá cả không quá một vài trăm nghìn USD tùy theo số lượng sản phẩm. Để có thể so sánh, hệ thống thiết bị quang-điện tử có chức năng tương tự nặng tới 250 kg (riêng khối quang-cơ) và giá bán hàng triệu USD.

b) Khả năng ứng dụng trong các hệ thống quang-điện tử để phát hiện và tự dẫn khí cụ bay

Hệ thống quang-điện tử để phát hiện và tự dẫn khí cụ bay (còn gọi là đầu tự dẫn quang-điện tử) về nguyên lí cũng giống như hệ thống quan sát và điều khiển hỏa lực nêu trên, nhất là khi ở chế độ sục sạo để phát hiện mục tiêu, yêu cầu hàng đầu là phạm vi quan sát (góc và cự li), còn chu kì cập nhật thông tin có thể chấp nhận là vài giây. Tuy nhiên do phải cung cấp thông tin liên tục để điều khiển bay trong giai đoạn tự dẫn khí cụ bay vào mục tiêu, cho nên sau khi phát hiện và “bắt” được mục tiêu hệ thống này cần phải cung cấp thông tin liên tục về mục tiêu đã chọn (với tần số không nhỏ hơn vài chục lần/s). Đặc biệt là trong chế độ này trường nhìn có thể chỉ cần vài độ đủ để khỏi bị “mất bám” mục tiêu.

Như vậy bằng phương pháp lựa chọn các chế độ điều khiển chuyển động cơ học tương ứng của camera ta có thể đạt được các yêu cầu của đầu tự dẫn quang-điện tử ở cả hai chế độ. Rõ ràng là với việc ứng dụng công nghệ xử lí ảnh nhanh ta vừa có thể mở rộng phạm vi quan sát ở chế độ sục sạo bằng cách tăng tốc độ quét của camera, vừa có thể tăng tốc độ cập nhật thông tin về mục tiêu ở chế độ bám sát mà vẫn đảm bảo cự li (do có khả năng xử lí ảnh với độ phân giải cao).

Khác với các thiết bị đặt ở mặt đất, đặt ở trên xe cơ động hoặc trên tàu chiến, ở đây ta xét khả năng ứng dụng trên các khí cụ bay (KCB) vì vậy yêu cầu rất quan trọng là phải có kích thước nhỏ gọn (không quá vài chục dm^3) trọng lượng nhẹ (không quá vài chục kg). Hai yêu cầu này chắc chắn sẽ đạt được khi ứng dụng công nghệ mạng nơron tế bào để chế tạo các thiết bị hệ thống quang-điện tử để phát hiện và tự dẫn KCB tầm trung (tầm bắn khoảng vài chục km đến vài trăm km) bởi vì các KCB này có kích thước và trọng lượng khá lớn, đồng thời có thể chấp nhận giá thành khá cao của đầu tự dẫn loại này.

c) Khả năng ứng dụng trong các hệ thống quan sát và phát hiện tội phạm

Ngày nay các phương tiện ghi hình (kể cả trong đêm tối) rất gọn nhẹ và giá cả không cao, cho nên các phương tiện này có thể đặt ở nhiều nơi và truyền theo các tuyến thông tin băng rộng

về các trung tâm xử lí. Như vậy ở các trung tâm này cần nhiều người túc trực ngày đêm để theo dõi, phát hiện các dấu hiệu nghi vấn, phát hiện các tên tội phạm đang cần truy nã và phát lệnh báo động khi cần thiết. Đặc điểm của công việc này là phải xử lí nhanh một lượng thông tin hình ảnh cực lớn. Nhiều khi cần phải hồi cứu rất nhiều băng đĩa ghi hình để điều tra một vụ án nào đó, chẳng khác nào “mò kim đáy biển”. Công nghệ xử lí ảnh nhanh có thể trợ giúp đắc lực cho các lực lượng an ninh trong những công việc này. Khi cần trợ giúp ở các trung tâm xử lí hình ảnh trong thời gian thực, công nghệ xử lí ảnh nhanh có thể phân tích sơ bộ và đánh dấu các đối tượng khả nghi theo một số dấu hiệu phân biệt : đàn ông-đàn bà, cao-thấp , già-trẻ, béo-gầy ... để trợ giúp các nhân viên an ninh tập trung chú ý vào các đối tượng cần thiết. Khi cần trợ giúp ở các trung tâm xử lí hình ảnh hồi cứu, công nghệ xử lí ảnh nhanh có thể phân tích sơ bộ và lọc ra các trường hợp có đối tượng khả nghi để giảm bớt (có thể đến hàng chục hàng trăm lần) khối lượng thông tin khổng lồ mà các nhân viên điều tra phải xem xét.

Ứng dụng công nghệ mạng nơ-ron tế bào để xử lí thông tin song song tốc độ cao

Như đã nêu ở trên khi cần tốc độ tính toán cao để giải các bài toán lớn bằng công nghệ mạng nơ-ron tế bào thì bài toán phải thuộc loại có thể đưa về dạng để giải bằng các thuật toán song song. Các bài toán giải *các phương trình đạo hàm riêng (trong cơ học thủy khí, trong dự báo thời tiết, ...)* có thể phục vụ gián tiếp cho quốc phòng-an ninh... Các bài toán mô phỏng tác chiến để huấn luyện chiến đấu và thử nghiệm các tình huống chiến đấu cũng có thể giải bằng các thuật toán song song. Do máy tính sử dụng công nghệ mạng nơ-ron tế bào có tốc độ tính toán rất cao (khoảng 12.10^{12} phép tính/giây) việc mô phỏng này có thể tiến hành rất tỷ mỉ trong thời gian thực cho đến từng người lính, từng khu vực chiến đấu với các điều kiện địa hình, thời tiết khác nhau trên cả chiến trường rộng lớn. Các chương trình mô phỏng này có thể trợ giúp rất nhiều cho việc huấn luyện các cấp chỉ huy và chiến sĩ sát với thực tế chiến đấu, đồng thời là phương tiện thử nghiệm để đánh giá hiệu quả các phương án tác chiến, các loại vũ khí mới...

1.7. Một số kết quả chính về nghiên cứu phát triển công nghệ CNN tại Viện MTA SzTAKI Hungary thời gian gần đây.

Viện sỹ Roska Tamás là một trong 2 nhà khoa học phát minh ra máy tính vạn năng tế bào CNN-UM trên thế giới. Phòng thí nghiệm về CNN của Viện sỹ Roska Tamás tại Viện MTA SzTAKI là một phòng thí nghiệm hàng đầu về CNN trên thế giới có đông đảo các giáo sư, TSKH và các cán bộ nghiên cứu trẻ làm việc. Phòng thí nghiệm về CNN tại Viện MTA SzTAKI là cơ sở hợp tác về CNN của nhiệm vụ nghị định thư này.

Với trên 15 năm kinh nghiệm nghiên cứu, phía đối tác đã xuất bản được nhiều sách chuyên khảo, công bố hàng trăm công trình trên các tạp chí và Hội nghị quốc tế, đào tạo hàng chục Tiến sỹ về CNN, mở nhiều khóa đào tạo về tính toán nơ ron ở các trường đại học Hungary và hợp tác với các cơ sở nghiên cứu ở các nước trên thế giới. Đặc biệt các sản phẩm về Bi-I và hệ phát triển Bi-I SDK của phía đối tác là các sản phẩm có tính đột phá và tiên phong về công nghệ CNN trên thế giới mở ra tiềm năng ứng dụng phong phú mà các công nghệ hiện hành chưa thực hiện được.

Một số kết quả khoa học chính về công nghệ CNN của phía đối tác Hungary trong thời gian gần đây được liệt kê như sau:

Các kết quả khoa học chính đạt được trong năm 2005-2006.

1. Thiết kế và thử nghiệm hoạt động của các siêu máy tính CNN có cấu trúc đặc thù trên FPGA như:
 - Mô hình mắt nhân tạo 12 lớp CNN thời gian thực,
 - Mô hình dòng chảy đại dương (ocean flow)
 - Mô hình vi phân đạo hàm riêng dạng Navier-Stokes
 - Mô hình sóng địa chấn.
2. Thiết kế và tạo chip thí nghiệm máy tính vạn năng CNN-UM có kiến trúc thích nghi cục bộ
3. Nghiên cứu cơ chế đồng bộ giữa các tế bào CNN dao động phi tuyến và khám phá ra loại sóng tròn ốc mới trên cơ sở mở rộng các kết nối của loại sóng khuếch tán.
4. Phát triển các thuật toán phát hiện và bám thích nghi đa mục tiêu di động qua xử lý đa kênh kết hợp với đặc tính động lực học của các đối tượng và thử nghiệm các thuật toán này trên máy tính xử lý ảnh Bi-I.
5. Phát triển các thuật toán phân tích và chú dẫn dòng ảnh video phức tạp.
6. Phát triển các thuật toán xử lý ảnh từ nhiều camera sử dụng thông tin mục tiêu, hình nền và nhiễu.
7. Phát triển các thuật toán mới cho nhận dạng các đặc tính sinh trắc học bao gồm nhận dạng mặt và các điệu dạng của tay.
8. Phát triển các thuật toán tránh va chạm mới cho đối tượng đặc thù và bài toán cảnh báo va chạm của các đối tượng chuyển động độc lập .
9. Phát triển các thuật toán phối hợp dạng sóng mới cho xử lý ảnh video hồng ngoại phối hợp với ảnh video thông thường.
10. Phát triển các thuật toán di truyền chạy trên CNN để nhận dạng đối tượng tốc độ cao tối ưu theo thuật di truyền.
11. Nghiên cứu các thuật toán xấp xỉ ngẫu nhiên và các kỹ thuật liên quan được ứng dụng trong phép phân tích tín hiệu đa nơ ron cho bài toán phát hiện sự kiện
12. Phát triển và thử nghiệm các thuật toán tựa đáp ứng miễn dịch cho các tác vụ phát hiện âm thanh và hình ảnh phức tạp.

13. Thử nghiệm kỹ thuật nâng cấp chất lượng hình ảnh đa mức dùng siêu máy tính CNN-UM (thời gian thực hiện với thuật nâng cấp ảnh 4 mức mà chỉ mất 3 ms trên hệ thị giác Bi-I)

Các kết quả khoa học chính đạt được trong năm 2004

1. Phát triển một thuật toán nền mới cho các bài toán xử lý đa cảm biến và định vị.
2. Thử nghiệm mô hình mắt nhân tạo dùng mạng CNN và phát triển mẫu mắt nhân tạo đơn giản và thử nghiệm trên 2 bệnh nhân ở Los Angeles với kỹ thuật cấy mắt (kết quả của sự cộng tác với trường đại học Harvard)
3. Phát triển các lớp ma trận trọng kết nối đồng bộ mới phụ thuộc cả không gian và thời gian cho các mạng CNN có đặc tính động lực dao động phi tuyến (kết quả của sự cộng tác với đại học U.C. Berkeley).
4. Phát triển hệ thống mã hóa quang học cả phần cứng/phần mềm trên máy tính xách tay quang học CNN (POAC)
5. Nghiên cứu về tính hỗn loạn (chaotic) trên chip CNN thị giác ACE 16k.
6. Phát triển các thuật toán tránh va chạm mới.
7. Phát triển thuật toán sinh trắc học mới cho quá trình nhận dạng người.

Các kết quả khoa học chính đạt được trong năm 2003

1. Các kết quả mới về lý thuyết:

- Đặt nền tảng cơ bản cho mô hình máy tính sóng tế bào xử lý chuỗi ảnh topographic.
- Thiết lập các nguyên lý cơ bản của các thuật toán CNN tựa hệ thống miễn dịch và phát triển các thuật toán cơ bản.
- Nghiên cứu các đặc tính của mạng CNN để tạo ra các mẫu chaotic (hỗn loạn) phụ thuộc không gian và thời gian
- Công bố các kết quả về tính ổn định mới của các mạng CNN nhiều lớp.
- Xác định các lớp thuật toán sóng và phát triển một số thuật toán di truyền học trên CNN

2. Các giải pháp cấu trúc và thuật toán mới.

- Thuật toán bám và định vị đa mục tiêu di động ứng dụng trong khảo sát địa hình và ứng dụng trong xử lý ảnh điện tim.
- Cấu trúc của thị giác nhân tạo CNN có khả năng điều chỉnh cảm biến thích nghi cục bộ.
- Phát triển cấu trúc máy tính analogic CNN quang điện tử khả trình (Programmable Optoelectronic Analogic CNN Computer – POAC) với thư viện (subroutines) và mẫu CNN quang học.
- Phát triển cấu trúc và các thuật toán nhúng trong môi trường giác quan đa trạng thái.
- Phát triển mảng cảm biến xúc giác và các thuật toán CNN
- Phát triển hệ nhận dạng mảng âm thanh spatial-temporal với các thuật toán CNN
- Thiết kế siêu máy tính CNN-UM trên cơ sở công nghệ FPGA và phát triển một số thuật toán dạng sóng.
- Các thuật toán CNN cho quá trình tính toán dòng quang học.

3. Các sản phẩm thiết bị sử dụng nguyên lý máy tính sóng CNN.

- Thiết kế và chế tạo đầu vào quang học analog và chip xử lý thị giác CNN-UM xử lý số theo từng tế bào.
- Thiết kế và chế tạo máy tính xách tay thị giác quang học POAC.

4. Các dự án hợp tác.

- Dự án Telesense cấp nhà nước (của Hungary: NKFP) bao gồm 9 cơ sở hợp tác phát triển công nghệ CNN. Trong đó các kết quả nổi bật như sau:

1- Hệ thị giác Bi-I, đã đoạt giải nhất sản phẩm năm của triển lãm Vision 2003 tại Stuttgart và là hệ thị giác có tốc độ nhanh nhất thế giới thời điểm đó.

2- Thí nghiệm kiểm tra đơn giản của hệ điện tim CNN với 1000 cơ sở dữ liệu echo-video.

3- Chíp cảm biến áp suất MEMS 3D

4- Nhiều kết quả nghiên cứu cơ bản về các mô hình cảm biến neuromorphic liên quan tới ứng dụng thị giác, ứng dụng về thính giác và cảnh báo.

- Dự án DICTAM (Dynamic Image Computing via Teraops Analogic Machines) trong hướng Kỹ thuật nổi trội tương lai của EU (FET- EU Future and Emerging Technology), với 5 đối tác quốc tế. Dự án đã hoàn thành xuất sắc với thử nghiệm thành công ở công ty Belgian. Dự án đã chứng minh sự thành công của mạch xử lý thị giác ACE 16k CNN và các thuật toán then chốt cho tác vụ nén video, cryptography và các ứng dụng kiểm tra thị giác.

- Tiến hành dự án nghiên cứu ONR cho mặt phẳng trọng tâm thích nghi CNN-UM dựa trên các chíp xử lý thị giác với Tây ban nha (trường đại học của Seville) và đối tác US (trường đại học Harvard) và tổng kết đánh giá kỹ thuật ở Budapest (đã thực hiện thành công).

- Bắt đầu dự án EU FET LOCUST cho công nghiệp ô tô. Dự án phối hợp năm đối tác quốc tế bao gồm nhóm nghiên cứu neurobiology và công ty VOLVO, mục tiêu cuối cùng là phát triển hệ thống an toàn ô tô qua hệ thống tránh nguy hiểm của Locust Insect, tránh va chạm và kỹ thuật phối hợp cảm biến qua các thuật toán tính sóng analogic CNN.

5. Xây dựng trung tâm nghiên cứu Info-Bionics Hungary

Thành lập Trung tâm nghiên cứu quốc gia chuyên ngành, lĩnh vực nổi bật về Info-Bionics, công năng của ngành điện tử, công nghệ thông tin và bio-Sciences (đặc biệt là trong lĩnh vực neurosciences). Nhân lực chính bao gồm các cán bộ nghiên cứu có học vị tiến sỹ (postdoc) tại 12 viện trên khắp toàn quốc, 6 phòng thí nghiệm nghiên cứu quốc gia trong Viện hàn lâm khoa học Hungary và 6 trường đại học.

Nhiệm vụ Nghị định thư may mắn được hợp tác trực tiếp với Viện sỹ Roska Tamás và nhóm nghiên cứu của Ông tại Viện MTA SzTAKI và đã nhận được nhiều hỗ trợ quý giá về tài liệu, phần mềm, về đào tạo, tham quan các thí nghiệm trực tiếp và thường xuyên được trao đổi, khoa học nhanh chóng qua e-mail và Internet. Sự hợp tác giúp đỡ nhiệt tình này đã rút ngắn được nhiều thời gian nghiên cứu trong nước. Khi tổ chức các hội thảo về CNN, đề tài được Viện sỹ Roska Tamás tư vấn về nội dung và hỗ trợ các tài liệu cần thiết và gửi bài cho Hội thảo về các kết quả nghiên cứu mới nhất trong lĩnh vực CNN.

2. XÂY DỰNG CÁC MÔ HÌNH VÀ THỬ NGHIỆM CÔNG NGHỆ XỬ LÝ ẢNH NHANH CNN PHỤC VỤ CHO NGHIÊN CỨU VÀ ĐÀO TẠO

Ngoài các nghiên cứu cơ bản tiếp cận cả về lý thuyết của mạng nơ ron tế bào CNN và công nghệ xử lý ảnh nhanh Bi-I nêu trên, nhiệm vụ nghị định thư còn đặt ra mục tiêu làm chủ và thử nghiệm công nghệ CNN trên một số mô hình trong phòng thí nghiệm phục vụ cho nghiên cứu và đào tạo. Phần báo cáo này mô tả các kết quả xây dựng các mô hình và thử nghiệm công nghệ CNN đã đạt được của nhiệm vụ nghị định thư.

2.1. Mô hình phát tia lửa điện phục vụ cho thí nghiệm thu ảnh tốc độ cao

2.1.1 XÂY DỰNG MÔ HÌNH

Giới thiệu chung

Thu ảnh tốc độ cao và phát hiện ảnh của sự kiện có thời gian tồn tại ngắn (Light Flicker Detection) là một yêu cầu thực tế cấp thiết của nhiều lĩnh vực trong công nghiệp, quốc phòng; như kiểm tra độ cách điện cho sứ ở điện áp cao, kiểm tra sự xuất hiện tia lửa điện khi đóng điện, phát hiện tia lửa đầu nòng của thiết bị hoả lực ...

Các hệ thống thị giác máy (machine vision) ngày càng được sử dụng rộng rãi trong công nghiệp trong việc kiểm tra, đảm bảo chất lượng và điều khiển quá trình cũng như các ứng dụng khác. Thông thường, các hệ thống như vậy bao gồm một hay nhiều camera, một card thu nhận ảnh, một máy tính và một bộ chương trình phần mềm xử lý ảnh. Tuy nhiên, trên thị trường cũng có các hệ camera thông minh tích hợp tất cả các bộ phận trên trong một thiết bị. Các thiết bị này có khả năng tự mình thực hiện một số nhiệm vụ ví dụ như phát hiện lỗi. Trong các thiết bị này, các thuật xử lý ảnh được thực hiện bởi một bộ vi xử lý DSP, hay như trong các hệ CNN đó là một chip xử lý ảnh CVM (Cellular Visual Microprocessor).

Hai vật cản lớn trong các quá trình tự động sản xuất và kiểm tra chất lượng sản phẩm là:

- Thuật toán xử lý ảnh đòi hỏi khối lượng tính toán rất lớn với tốc độ xử lý nhanh mà các máy tính chưa đáp ứng được.
- Các yêu cầu kiểm tra thông thường đòi hỏi một tần số lấy mẫu cao hơn rất nhiều lần so với các sensor và camera có khả năng.

Các rào cản này đã được giải quyết triệt để với sự ra đời của chip xử lý ảnh CVM trên công nghệ CNN. Chip CVM có khả năng thu thập và xử lý hàng nghìn ảnh trong một giây trong thời gian thực trong khi các hệ thống ảnh số thông thường chỉ có thể xử lý với tần số một hoặc hai trăm ảnh trong một giây.

Về bản chất, CVM là một mảng cảm biến ảnh dựa trên kỹ thuật CMOS tích hợp với một hệ xử lý ảnh. Hệ xử lý ảnh dựa trên công nghệ CNN xử lý ảnh đa bộ vi xử lý và bao gồm nhiều phần tử vi xử lý đồng dạng. Khi thực hiện một thao tác trên ảnh, một bộ vi xử lý thông thường thực hiện thao tác trên từng điểm ảnh. Trong bộ đa vi xử lý của CVM, tất cả điểm ảnh của một ảnh được xử lý bởi các phần tử vi xử lý khác nhau nên thao tác trên ảnh có thể được thực hiện đồng thời cho tất cả các điểm ảnh. Phần tử xử lý được kết nối trực tiếp tới các phần tử lân cận cho phép tạo ra các thao tác lân cận tốc độ cao.

Với cấu trúc song song và kết nối lân cận, CVM có thể thực hiện các thao tác trên toàn bộ ảnh trong một vài milli giây. Khả năng tính toán của CVM tương đương với 10^{12} phép tính

trong một giây. CVM cũng có thể lưu các chương trình như các máy tính thường và có thể được lập trình từ các môi trường phát triển xử dụng ngôn ngữ lập trình chuyên dụng hoặc quen thuộc như C và C++.

Để thử nghiệm xây dựng các phần mềm theo công nghệ CNN thu được ảnh tốc độ cao, cũng như thử nghiệm các tình huống xuất hiện ánh sáng có thời gian tồn tại ngắn cần có một thiết bị đơn giản sử dụng trong phòng thí nghiệm, đáp ứng được các yêu cầu của một số tình huống tiêu biểu trong thực tế.

Trong mô hình này, khả năng xử lý tốc độ cao của CVM được dùng để phát hiện và phân loại các tia lửa điện phóng ra từ đầu bugi trong thời gian vài milli giây (Hình 2.1). Độ phân giải cao trong miền thời gian cũng tạo ra khả năng theo dõi và kiểm tra quá trình hình thành của tia lửa. Đồng thời thiết bị cũng cần có kết cấu nhỏ gọn, tiêu thụ ít năng lượng, an toàn, sử dụng được nguồn điện dân dụng thông dụng



Hình 2.1 Hình ảnh bugi và tia lửa điện

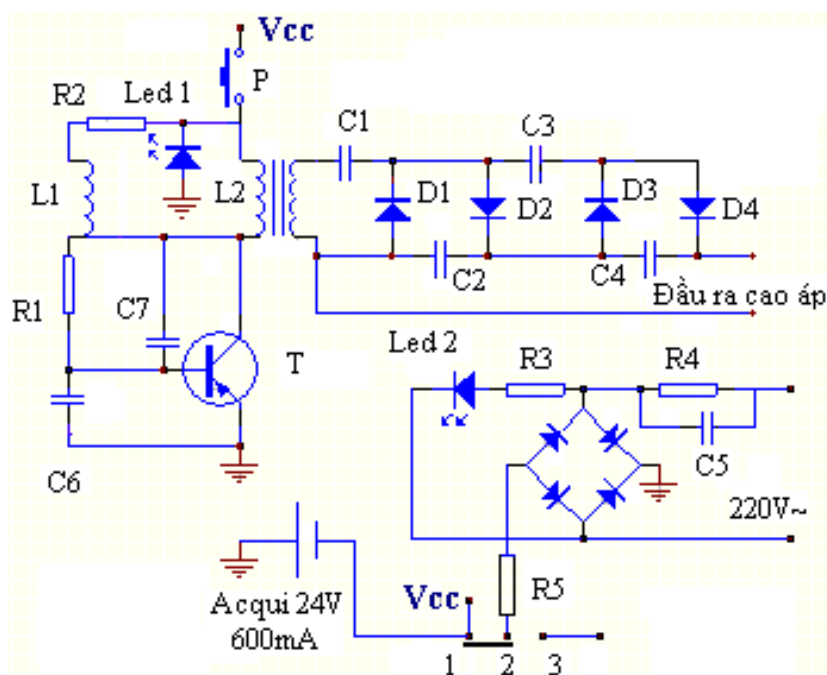
Yêu cầu làm việc, đặc tính kỹ thuật của thiết bị

Thiết bị được chế tạo có các đặc tính kỹ thuật chính như sau:

- + Phát tia lửa điện qua cực bugi với chiều dài tia lửa điện lớn nhất 3mm
- + Tốc độ phát tia lửa điện khi khe hở bugi 1.5mm đạt 80 lần/phút
- + Kích thước 150 x 140 x 85
- + Trọng lượng 400g
- + Nguồn điện nạp 220VAC 50/60Hz

Cấu tạo và nguyên lý hoạt động

Sơ đồ mạch điện



Hình 2.2. Mạch nguyên lý của thiết bị phát tia lửa điện

Khi nạp điện: công tắc điều khiển (Hình 2.2) được gạt xuống, nối các tiếp điểm 1-2. Điện áp 220V~ qua R4 và C5 sụt áp qua mạch cầu nắn còn khoảng 30VDC nạp cho Acqui. Điện trở R5 để điều chỉnh dòng nạp.

Khi phát tia lửa điện: Công tắc được gạt về vị trí nối 2 với 3. Lúc này nếu ta ấn nút P tranzistor được cấp điện, mạch dao động Blocking sẽ làm việc. Điện áp ở đầu ra thứ cấp của biến áp xung được nắn bội áp qua các tụ từ C1..C4 và các diot D1..D4 sẽ cho ra điện áp 1500-1700V. Điện áp này đưa vào các cực của bugi để phóng điện.

Giá trị các linh kiện sử dụng:

$$C1=C2=C3=C4=0.0022/2kV$$

D1, D2, D3, D4 diot có điện áp ngược $\geq 2kV$

$$R4 = 1M\Omega, R5 = 330\Omega$$

$$C5 = 0.33/250V\sim$$

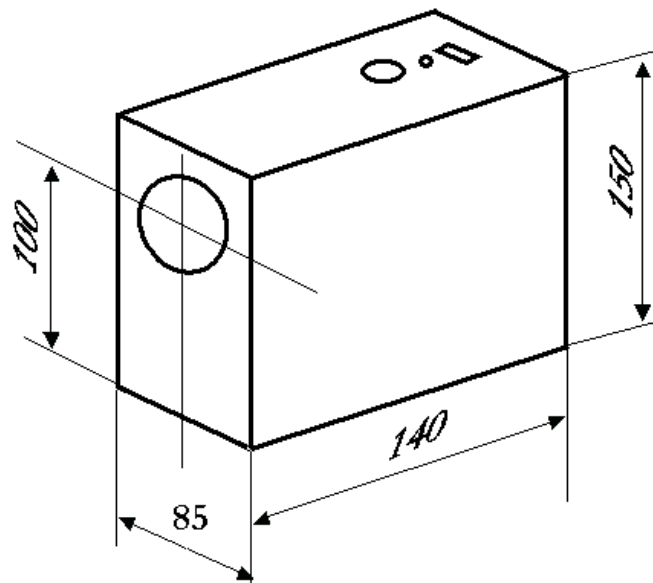
Acqui 24V 600mA

Tranzixtor tương đương H1061

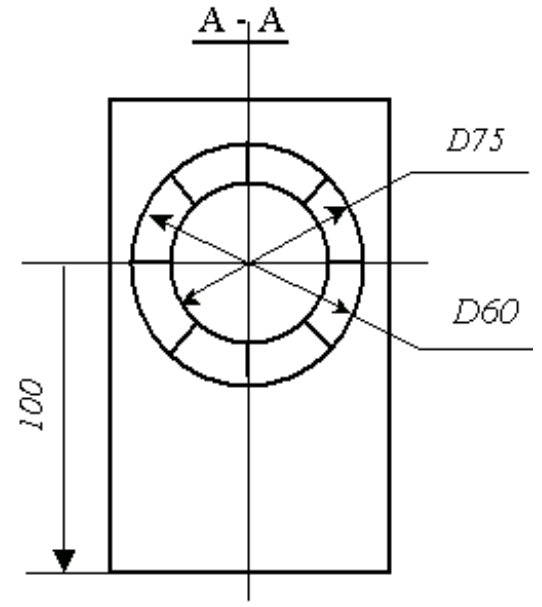
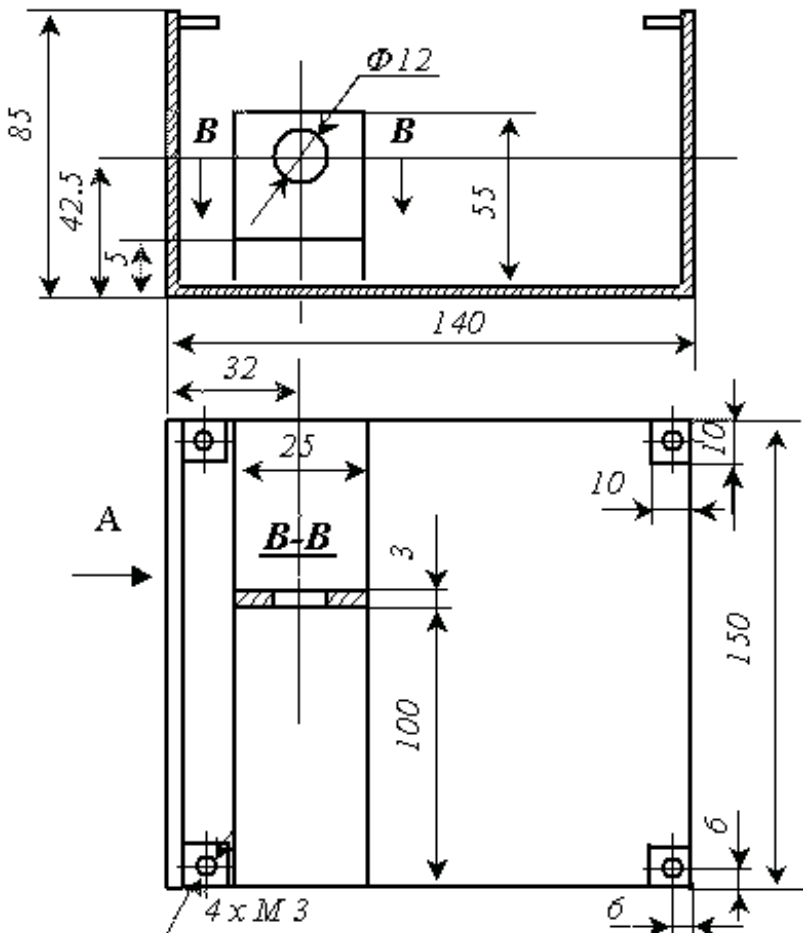
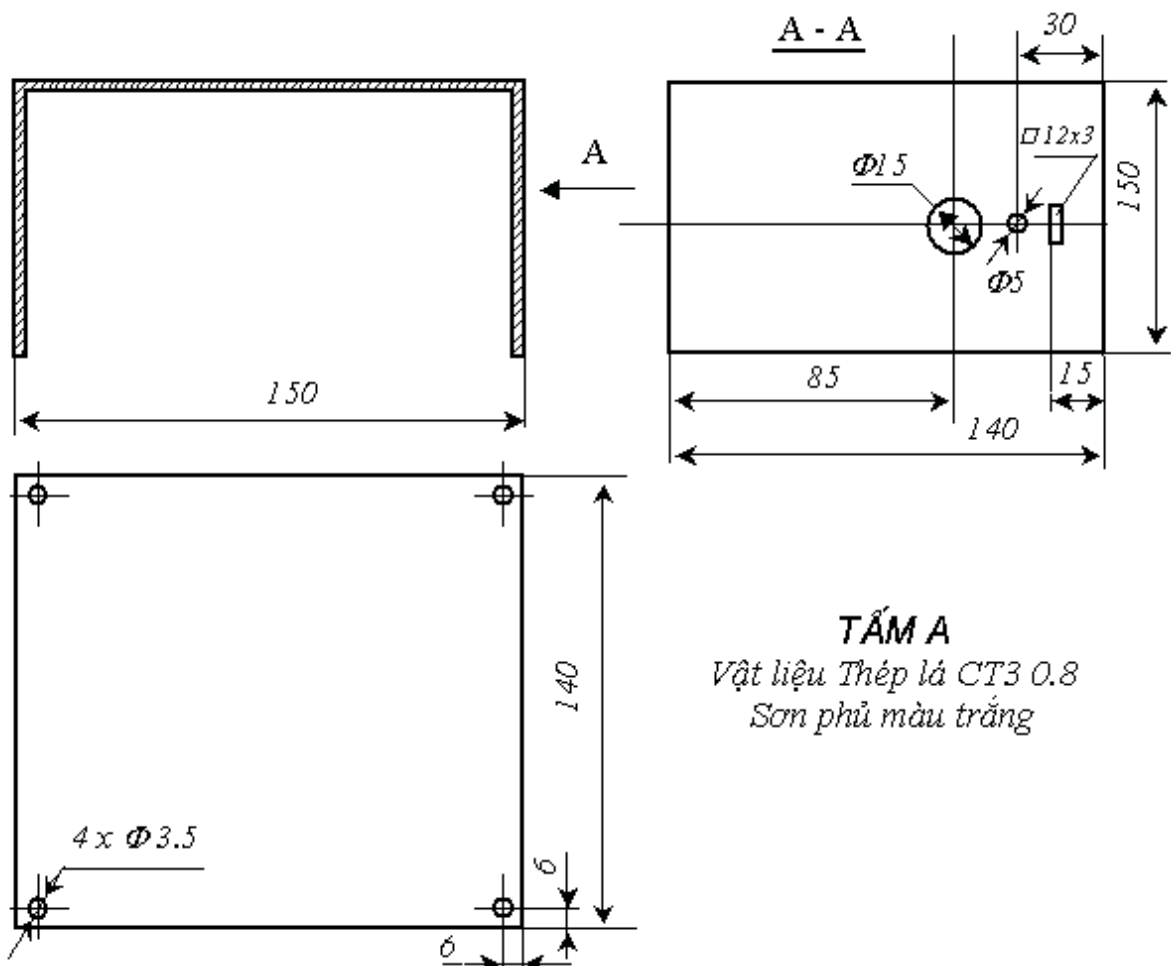
Mạch cầu nắn điện 400V~/1A

Kết cấu vỏ

Vỏ thiết bị sử dụng tôn 0.8mm gồm 2 nửa. Bảng mạch điện được gắn lên lên một nửa vỏ A; hai nửa vỏ A và B được gắn với nhau bằng 4 vít M3.



Hình 2.3. Kích thước vỏ thiết bị phát tia lửa điện



TẤM B
 Vật liệu Thép lá CT3 0.8
 Sơn phủ màu trắng

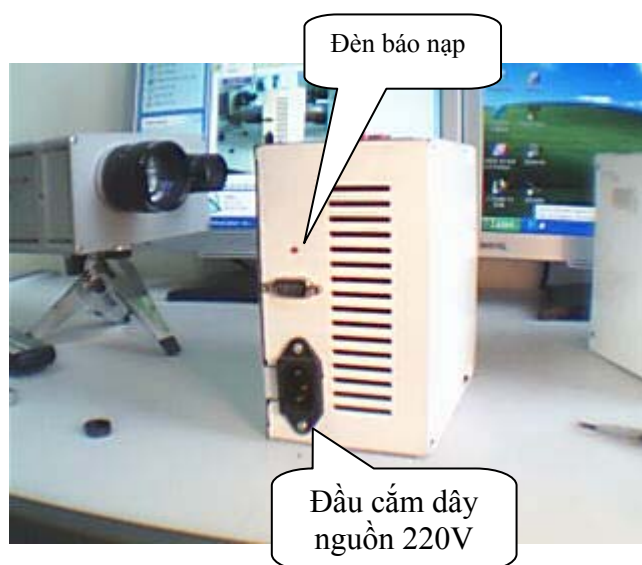
Hình 2.4. Các tấm A và B



Hình 2.5. Bộ phát tia lửa điện

Nạp điện

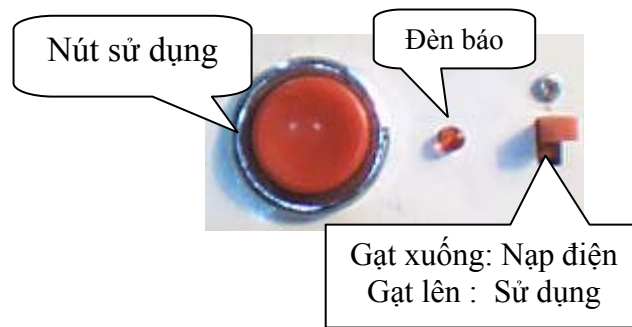
Việc nạp điện cho Acqui của thiết bị sử dụng nguồn điện 220VAC. Bật công tắc điều khiển về vị trí nạp điện (Hình 2.7). Dùng dây nối loại cắm nguồn máy tính tiêu chuẩn nối vào "Đầu cắm dây nguồn 220V" ở phía sau thiết bị (xem hình 2.6.). Cắm vào ổ cắm 220V~. Quan sát đèn báo nạp sáng nếu hoạt động bình thường. Sau khoảng 5 phút nạp đã có thể sử dụng được. Quá trình nạp đầy đủ 30 phút.



Hình 2.6. Thiết bị phát tia lửa điện nhìn từ phía sau

Phát tia lửa điện

- Khi nạp: Công tắc được gạt xuống dưới như hình 2.7. Sau khi đã nạp điện cho Acqui của thiết bị bằng nguồn điện 220 V ~ ta có thể rút dây nạp ra để mang thiết bị đi sử dụng di động.
- Khi phát tia lửa điện: Gạt công tắc lên trên như hình 2.7. Lúc này nếu ta bấm nút sử dụng tia lửa điện sẽ được phát ra qua cực của bugi. Đèn báo sẽ bật sáng thể hiện thiết bị đang hoạt động.



Hình 2.7. Các nút điều khiển



Hình 2.8. Đầu bu gi bật tia lửa điện

Bảo quản thiết bị

- + Bảo quản thiết bị ở nhiệt độ trong phòng
- + Tránh nơi ẩm ướt, tránh nước hoặc chất lỏng chảy vào thiết bị
- + Khi không sử dụng gạt công tắc sang vị trí nạp điện (dù cho có nạp điện hay không)
- + Nếu thời gian lâu không sử dụng khoảng 2 tháng nạp lại Acqui cho thiết bị một lần, thời gian khoảng 30 phút để tránh hỏng Acqui

Tích hợp mô hình thí nghiệm thu ảnh tia lửa điện

Hệ thống được tích hợp bao gồm:

- Máy tính Pentium 2.4GHz
- Bi-I camera kết nối mạng Ethernet 10/100 MHz
 - . ACE16K Processor
 - . Embedded TI C6415
- Thiết bị tạo tia lửa điện
- Hệ phát triển Bi-I: Bi-I Run; Bi-I Config
- ACE Editor
- Chương trình dịch ngôn ngữ sử lý song song cho chip ACE16K
- Chương trình dịch Code Composer cho DSP
- Phần mềm thu và xử lý ảnh tia lửa điện

2.1.2 THỬ NGHIỆM PHÁT HIỆN VÀ PHÂN TÍCH TIA LỬA ĐIỆN VỚI TỐC ĐỘ CAO TRONG THỜI GIAN THỰC.

Với mô hình nêu trên tia lửa được phát ra liên tục khi được cấp điện và độc lập so với phần sensor và vi xử lý. Kích thước thực của điểm ảnh vào khoảng $200 \mu\text{m} \times 200 \mu\text{m}$. Sau khi hoàn tất quá trình chỉnh tiêu cự, hình ảnh cho thấy rằng các tia lửa có kích thước hẹp hơn kích thước điểm ảnh. Do đó, độ rộng của tia lửa thông thường là một điểm ảnh.

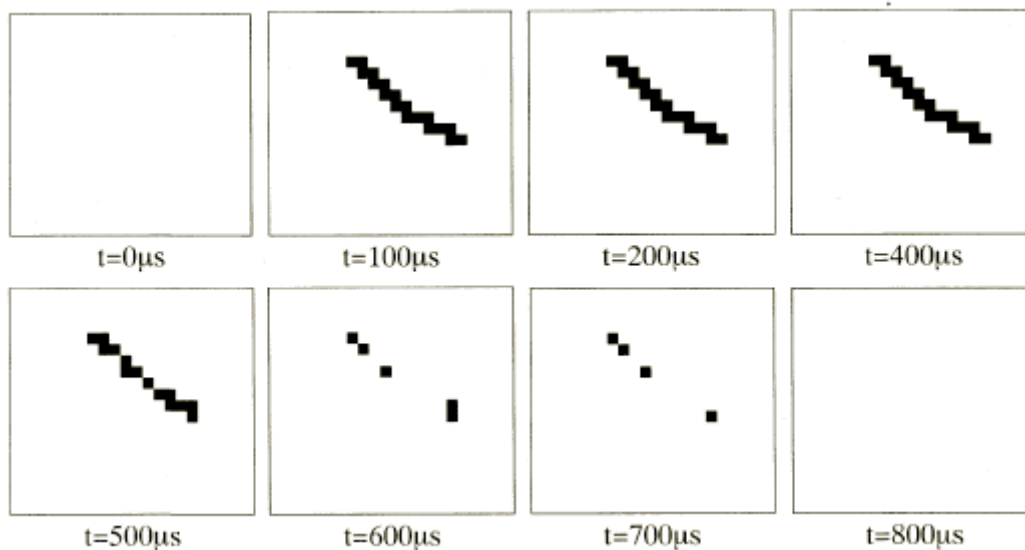
Sử dụng bộ camera Bi-I V2 với chip CVM là ACE16KV2, hệ thống có khả năng thu thập ảnh và lưu vào 8 bộ nhớ trong camera ở tốc độ 500000 ảnh/giây. Còn nếu quá trình phân loại được tiến hành thì tốc độ xử lý tối đa là trên 30000 ảnh/giây.

Thời gian hình thành cho một tia lửa điện vào khoảng $700 \mu\text{s}$ và thời gian giữa hai tia lửa liên tiếp vào khoảng 5-6 ms. Điều này có nghĩa là chúng ta có thể ghi lại khoảng 7 ảnh cho một tia lửa điện với hệ thống chạy tại tốc độ 10000 ảnh/giây.

Hình 2.9 cho thấy một chuỗi ảnh thu được từ cùng một tia lửa điện. Các hình cách nhau một khoảng thời gian là $100 \mu\text{s}$. Tia lửa điện xuất hiện gần như tức thời. Sau đó, nó ổn định không thay đổi vị trí và hình dạng trong khoảng thời gian khoảng $200 \mu\text{s}$ và cuối cùng từ từ mất đi.

Phần mềm thí nghiệm SPARKS do nhiệm vụ nghị định thư nghiên cứu phát triển dựa trên thư viện các hàm xử lý ảnh nhanh của công nghệ CNN được thực hiện bởi bộ xử lý ACE16K cung cấp các khả năng sau:

- Phát hiện tia lửa điện
- Xây dựng ảnh hợp của một chuỗi các tia lửa
- Phân tích ảnh hình thái học
- Nhận dạng các tia lửa chẻ
- Theo dõi vị trí của các tia lửa
- Phân tích thống kê về vị trí, độ chẻ và khoảng thời gian
- Ước tính năng lượng của tia lửa



Hình 2.9 Chuỗi ảnh từ một tia lửa điện với thời gian giữa hai ảnh là $100 \mu\text{s}$

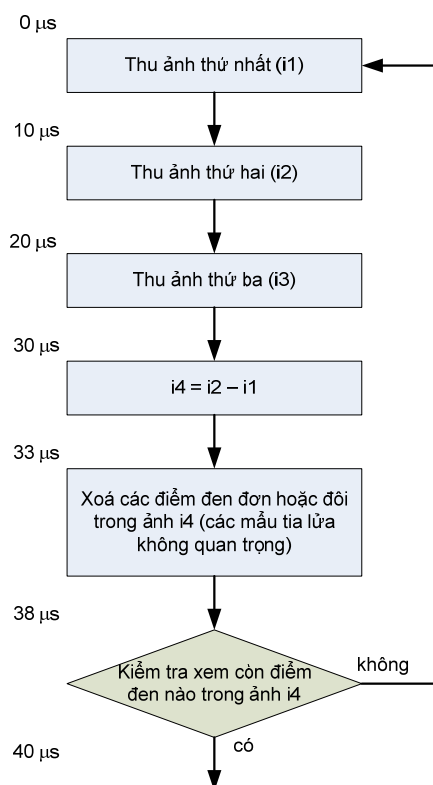
Một số kết quả thí nghiệm đạt được

Phát hiện sự xuất hiện của tia lửa

Như mô tả ở trên, tia lửa hình thành một cách đột ngột từ giữa 2 ảnh liên tiếp và sau đó từ từ mất đi. Với khả năng xử lý ảnh nhanh của camera, chúng ta có thể kiểm tra sự hình thành của các tia lửa điện với tần số lấy mẫu trên 50kHz.

Ba ảnh liên tiếp được thu thập và lưu vào bộ nhớ bên trong của camera. Sự hình thành của tia lửa sẽ được phát hiện thông qua việc so sánh ảnh thứ nhất và thứ nhì. Ảnh thứ nhất không chứa tia lửa nhưng trong ảnh thứ hai thì một phần đáng kể của tia lửa đã được tạo thành.

Lưu đồ của thuật toán phát hiện tia lửa điện được mô tả như trong Hình 2.10. Thuật toán bao gồm một phép trừ ảnh (image subtraction), một phép ăn mòn (erosion) và một hàm hoặc (OR) cho toàn bộ ảnh. Tất cả các hàm này đều có trong bộ vi xử lý của camera. Thời gian thực hiện khoảng 40 μ s. Hình 2.11 cho thấy quá trình hình thành của tia lửa điện.



Hình 2.10: Lưu đồ thuật toán phát hiện sự hình thành tia lửa

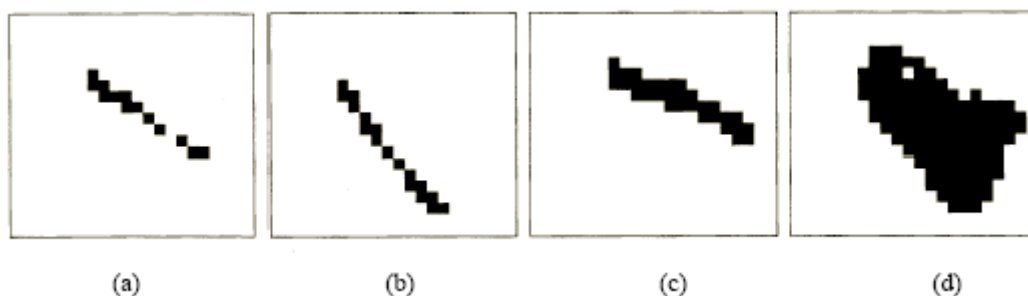


Hình 2.11 Sự hình thành của tia lửa

Xây dựng ảnh hợp của các chuỗi tia lửa

Trong thí nghiệm này, chúng ta thu thập một số lượng lớn ảnh tia lửa và hợp chúng với nhau. Bức ảnh hợp này sẽ cho chúng ta biết các tia lửa có nằm ở trong vị trí phù hợp hay không. Mặc dù phép hợp ảnh này đơn giản, nó rất quan trọng trong công nghiệp chế tạo bugi vì nếu bộ phận cách ly bằng gốm chứa một vết nứt nhỏ, một phần của tia lửa sẽ đi qua khe nứt này thay vì ở vị trí thông thường. Nếu tia lửa không tập trung ở đúng vị trí, bộ đánh lửa sẽ không hoạt động bình thường. Các bugi này cần được phát hiện và loại bỏ. Tuy nhiên, nếu lượng tia lửa rò rỉ ít, chúng ta cần thử nhiều lần (thông thường là 1000 lần hay hơn) để tìm ra bugi có lỗi. Nếu một dây chuyền sản xuất tự động tạo ra 1 hoặc 2 cái bugi trong một giây, việc kiểm định chất lượng trở nên vô cùng khó khăn cho các camera thông thường do thời gian eo hẹp. Công việc này lại rất đơn giản đối với công nghệ CVM.

Công việc hợp ảnh là một thao tác đơn của các bộ đa vi xử lý tiêu cự phẳng và chỉ tốn khoảng vài μ s. Việc thu ảnh và hợp chúng mất khoảng 40 μ s và cho phép hệ thống chạy với tốc độ 25000 ảnh/giây. Thậm chí quá trình sản xuất bugi còn có thể được tăng lên mà việc kiểm tra vẫn có thể được đảm bảo chính xác.

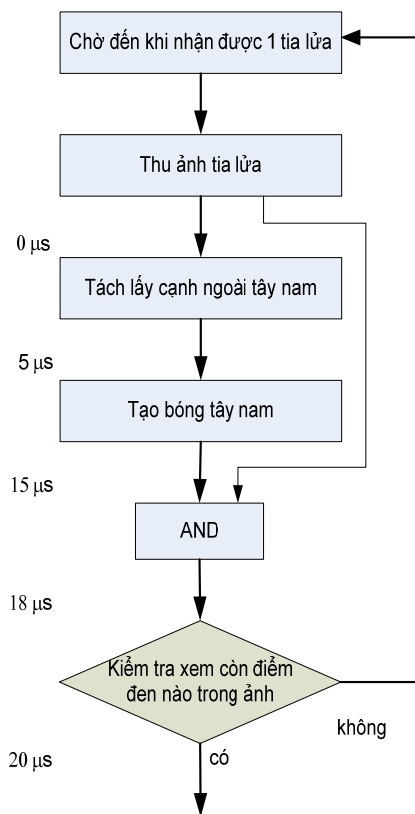


Hình 2.12: Hợp ảnh của chuỗi các tia lửa: (a) tia lửa ở vị trí bình thường; (b) tia lửa ở vị trí thấp; (c) tia lửa ở vị trí cao; (d) ảnh hợp của 1000 tia lửa

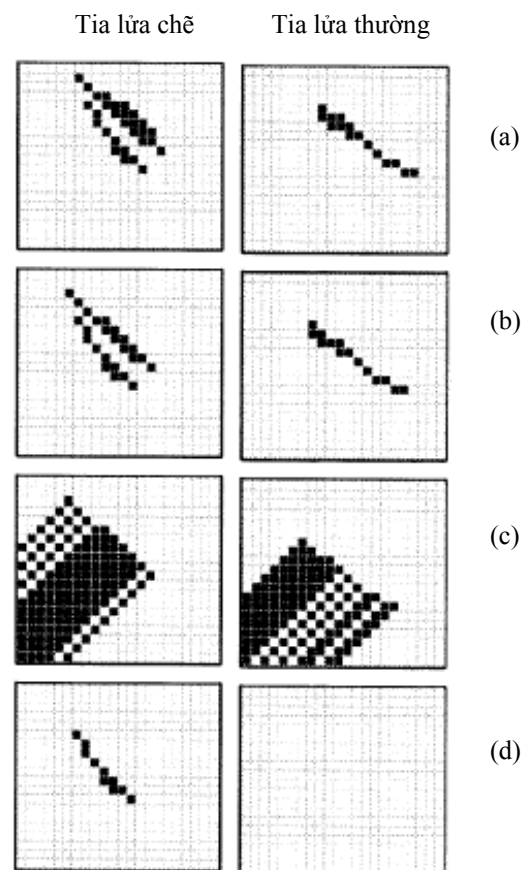
Nhận dạng các tia lửa chệch

Thí nghiệm này tập trung vào việc tìm kiếm các tia lửa hiếm. Chúng ta biết rằng một phần rất nhỏ (1 trong số 5000) trong số các tia lửa phóng ra ngoài không khí. Một thuật toán được phát triển để phát hiện các tia lửa có chứa 2 nhánh. Thuật toán sẽ tạo ra phần bóng về hướng tây nam của tia lửa. Nếu đây là tia lửa đơn, phần bóng và phần tia lửa sẽ không giao nhau trên ảnh. Tuy nhiên, nếu tia lửa là tia lửa chệch, phần bóng sẽ che phủ lên nhánh nam của tia lửa.

Lưu đồ của thuật toán được mô tả trên Hình 2.13. Thời gian để thực hiện thao tác tìm kiếm tia chệch vào khoảng 20 μs . Vì tia lửa thường tồn tại trong khoảng thời gian là 500 μs , việc phát hiện tia chệch có độ ổn định cao. Lưu đồ thuật toán được mô tả trong Hình 2.13. Một số kết quả thí nghiệm được thể hiện ở Hình 2.14.



Hình 2.13: Lưu đồ thuật toán tìm tia lửa chệch



Hình 2.14: Kết quả các bước tìm tia lửa chệch (a) ảnh tia lửa; (b) tách cạnh hướng tây nam; (c) tạo bóng hướng tây nam; (d) AND ảnh bóng và ảnh tia lửa ban đầu

2.1.3 HƯỚNG DẪN SỬ DỤNG THÍ NGHIỆM TIA LỬA ĐIỆN SPARKS

Giới thiệu chung

Nhiệm vụ Nghị định thư đã xây dựng mô hình thí nghiệm thu ảnh tia lửa điện bao gồm các thiết bị phần cứng và các phần mềm thí nghiệm SPARKS để phục vụ cho nghiên cứu và đào tạo công nghệ CNN trong phòng thí nghiệm. Thí nghiệm tia lửa điện nhằm mô tả khả năng phát hiện, thu nhận và xử lý tốc độ cao của công nghệ CNN được dùng để phát hiện và phân loại các tia lửa điện phóng ra từ đầu bugi trong thời gian vài milli giây. Độ phân giải cao trong miền thời gian cũng tạo ra khả năng theo dõi và kiểm tra quá trình hình thành của tia lửa.

Phần mềm thử nghiệm kiểm tra nhanh tia lửa điện SPARKS đã được phát triển và nạp trong đĩa CD “ *các mô hình và chương trình thử nghiệm công nghệ xử lý ảnh tốc độ cao phục vụ cho nghiên cứu và đào tạo*” của nhiệm vụ nghị định thư.

Mô hình thí nghiệm bao gồm 2 phần chính. Phần thứ nhất là một sensor ảnh có tiêu cự phẳng cùng một bộ vi xử lý và các phần cứng-phần mềm đi kèm (camera Bi-I và hệ phát triển phần mềm CNN). Phần thứ hai là một bộ tạo tia lửa điện (bugi). Tia lửa được phát ra liên tục khi được cấp điện và độc lập so với phần sensor và vi xử lý.

Kích thước thực của điểm ảnh vào khoảng $200 \mu\text{m} \times 200 \mu\text{m}$. Tuy nhiên, khi lên ảnh, các tia lửa thường tạo ra các ảnh nhoè do ánh sáng của tia lửa phát tán ra xung quanh. Thông qua việc hiệu chỉnh tiêu cự và khoảng cách giữa camera và bộ tạo tia lửa điện và với việc sử dụng các hàm sketon sẽ giúp trích lọc ra được ảnh tia lửa thích hợp.

Sử dụng bộ camera Bi-I V2 với chip CVM là ACE16KV2, hệ thống có khả năng thu thập ảnh và lưu vào 8 bộ nhớ trong camera ở tốc độ 500000 ảnh/giây. Còn nếu quá trình phân loại được tiến hành thì tốc độ xử lý tối đa là trên 30000 ảnh/giây.

Thời gian hình thành cho một tia lửa điện vào khoảng 700 μs và thời gian giữa hai tia lửa liên tiếp vào khoảng 5-6 ms.

Các bước thực hiện 1 thí nghiệm tia lửa điện

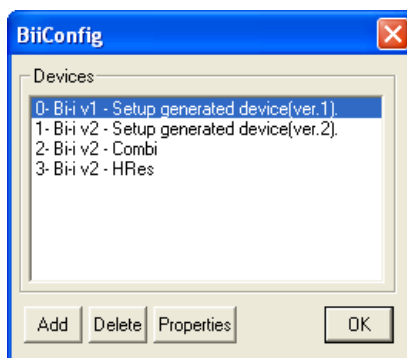
Một số thí nghiệm khác nhau nhằm phát hiện, đo đạc một số đặc tính của tia lửa điện đã được tiến hành.

Với từng thí nghiệm thuật toán thu thập và xử lý được xây dựng trên môi trường Code Composer Studio 3.1, của Texas Instruments. Các chương trình này được biên dịch ra file có đuôi .OUT thông thường nằm trong thư mục Output là dạng mã máy của camera CNN.

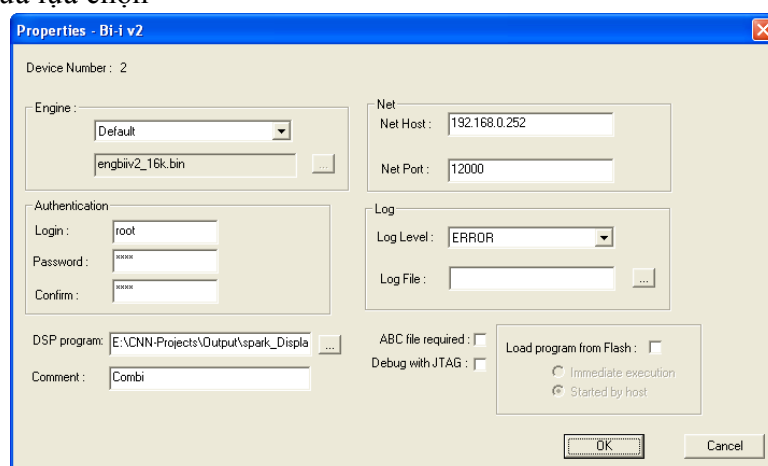
Để chạy các thí nghiệm này, các bước tiến hành như sau:

Bước 1: Nạp file .OUT xuống camera

- Mở chương trình **BiiConfig** trong menu **Bi-i 1.10 Development** để đặt cấu hình cho Camera. Màn hình sau sẽ xuất hiện cho phép người dùng thêm bớt các camera CNN có trên mạng.



Chọn camera sử dụng cho thí nghiệm tia lửa điện và ấn Properties. Màn hình sau sẽ hiện ra để cho người dùng đặt các thông số cho camera CNN vừa lựa chọn

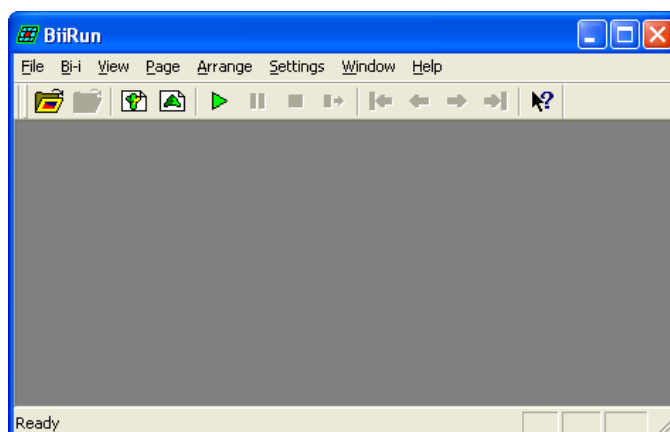


Người dùng nhập chính xác các thông số cho camera đặc biệt là địa chỉ IP của camera (Net Host) và username/password. Trong mục *DSP program* lựa chọn file .OUT của thí nghiệm cần thử.

Lưu ý: cần xoá mục *ABC file required*.

Sau đó ấn OK và có thể đóng phần mềm **BiiConfig**.

- Mở chương trình **BiiRun** trong cùng menu **Bi-i 1.10 Development**. Đây sẽ là phần giao diện giữa người sử dụng và camera CNN thông qua kết nối Ethernet.



Trong menu của chương trình này, chọn mục *Settings/Run*. Chọn đúng camera CNN vừa được cấu hình ở phần trên rồi ấn OK.

Bước 2: Kiểm tra kết nối giữa camera CNN và máy tính

Việc kiểm tra có thể thông qua sử dụng lệnh **ping** tới địa chỉ IP của camera CNN; hoặc dùng *Internet Explorer* và gõ địa chỉ IP của camera CNN, camera CNN có hỗ trợ Web Server để dùng cho cấu hình camera từ xa. Khi kiểm tra xong thấy kết nối được, đóng các chương trình dùng để thử kết nối lại.

Bước 3: Thực hiện chương trình

Chương trình thí nghiệm được thực hiện thông qua lệnh *Bi-i/Run* ở trên menu của chương trình **BiiRun** (hoặc ấn Ctrl-R). Các màn hình giao tiếp do chương trình tự động tạo ra sẽ được mở.

Các thí nghiệm với tia lửa điện

a. Thí nghiệm 1:

Mục đích: Làm quen với camera CNN.

File .OUT: spark_Display64_Intro.OUT

Mô tả: Thí nghiệm mô tả khả năng phát hiện và tách ảnh của tia lửa một cách liên tục và nhanh chóng với tốc độ bắt ảnh thông thường.

b. Thí nghiệm 2:

Mục đích: Giới thiệu khả năng bắt ảnh nhanh của camera CNN.

File .OUT: spark_Display64_HighAcq.OUT

Mô tả: Thí nghiệm mô tả khả năng phát hiện và tách ảnh của tia lửa một cách liên tục và nhanh chóng với tốc độ bắt ảnh trên 50000 ảnh/giây.

c. Thí nghiệm 3:

Mục đích: Giới thiệu khả năng xử lý các tính toán ảnh nhanh của camera CNN.

File .OUT: spark_Display64_Display.OUT

Mô tả: Thí nghiệm mô tả khả năng phát hiện và tách ảnh của tia lửa một cách liên tục và nhanh chóng với tốc độ bắt ảnh trên 50000 ảnh/giây. Đồng thời phần xương của tia lửa (footprint) được tách ra cho ta thấy rõ hình dạng của tia lửa.

d. Thí nghiệm 4:

Mục đích: Giới thiệu khả năng xử lý các tính toán ảnh nhanh của camera CNN.

File .OUT: spark_Display64_Size.OUT

Mô tả: Thí nghiệm mô tả khả năng phát hiện và tách ảnh của tia lửa một cách liên tục và nhanh chóng với tốc độ bắt ảnh trên 50000 ảnh/giây. Các tia lửa phát hiện được sẽ được phân loại dựa trên kích thước thành 2 loại (to và nhỏ).

Cùng với việc sử dụng hệ thống camera CNN (BI-I) thí nghiệm khảo sát phân tích tia lửa điện đã mô tả khả năng thu thập và phân tích ảnh của các tia lửa phát ra từ một bugi ở tốc độ thu thập và xử nhanh (30000 ảnh/giây). Quá trình này có thể được áp dụng vào dây chuyền sản xuất và kiểm tra chất lượng bugi trong công nghiệp. Khả năng xử lý ảnh vô cùng nhanh, chính xác và liên tục này của hệ thống CVM có thể được thay đổi một cách dễ dàng và áp dụng cho các ứng dụng tương tự khác.

2.2. Mô hình nhận dạng kiểm tra sản phẩm tốc độ cao phục vụ cho nghiên cứu và đào tạo

2.2.1 XÂY DỰNG MÔ HÌNH

Yêu cầu làm việc, đặc tính kỹ thuật của thiết bị

Trong các thí nghiệm xử lý ảnh tốc độ cao thường có yêu cầu thu ảnh của đối tượng chuyển động nhanh tới hàng trăm km/h, thời gian tồn tại trước ống kính rất ngắn. Ta có thể sử dụng thiết bị đĩa quay để thay thế cho chuyển động thẳng. Khi đó tốc độ dài $= \pi D/60$. Với một đĩa $D=150\text{mm}$ gắn một đối tượng quay với 3000vg/ph có thể tạo ra tốc độ khoảng 85 km/h ; nếu gắn hai đối tượng đối xứng có thể tạo ra đối tượng chuyển động 170km/h Điều chỉnh vô cấp tốc độ motor trục đĩa sẽ điều chỉnh được vô cấp tốc độ chuyển động của đối tượng. Do vậy dù có cấu tạo đơn giản nhưng thiết bị đĩa quay là một thiết bị rất cần thiết cho các thí nghiệm thu, phân tích, xử lý ảnh tốc độ cao dùng công nghệ CNN.

Đặc tính kỹ thuật của thiết bị được chế tạo:

- + Tốc độ quay nhỏ nhất của trục đĩa 100vg/ph
- + Tốc độ quay lớn nhất của trục đĩa 3000vg/ph
- + Đường kính đĩa lớn nhất lắp được trên thiết bị 150 mm
- + Tốc độ quay điều khiển vô cấp từ min \rightarrow max và từ max \rightarrow min
- + Kích thước $260 \times 230 \times 150$
- + Trọng lượng 1200g
- + Nguồn điện sử dụng 220VAC

Cấu tạo và nguyên lý hoạt động

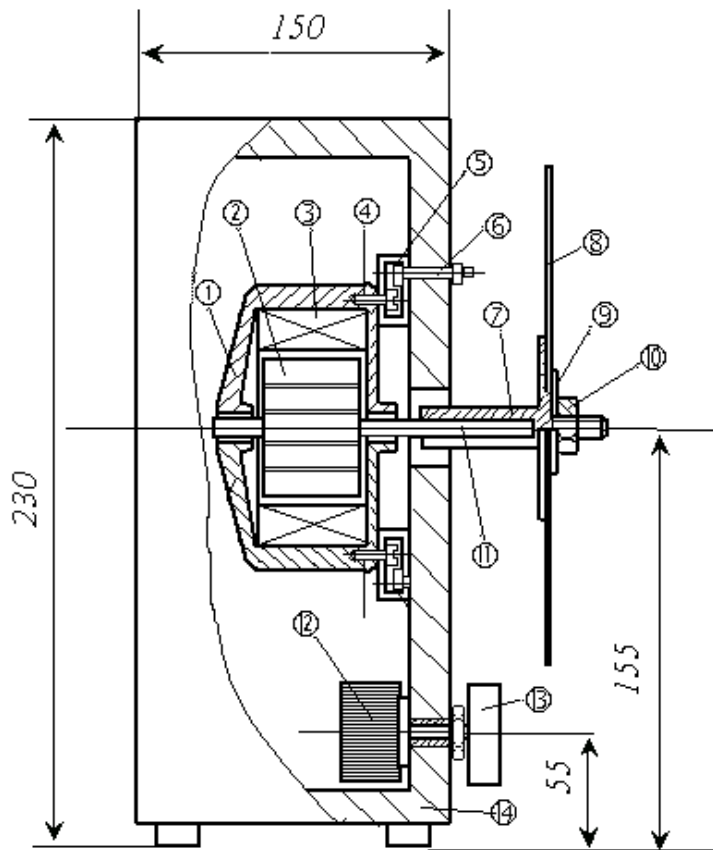
Thiết bị bao gồm một động cơ AC rô to lồng sóc công suất 35W gắn với một trục cơ khí có gờ và ốc giữ để gắn được đĩa có mang ảnh của đối tượng trên đĩa. Điều chỉnh tốc độ động cơ bằng cách thay đổi điện trở nối tiếp $20\Omega/40\text{W}$ với cuộn dây stator của động cơ.



Hình 2.15: Thiết bị quay

Kết cấu cơ khí

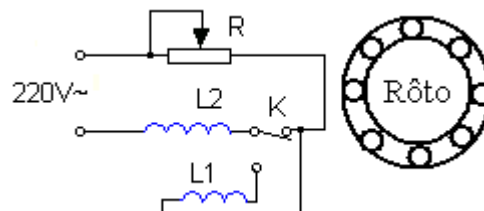
Động cơ được gắn trên đế nhôm cố định với vỏ thiết bị. Vỏ thiết bị sử dụng vật liệu gỗ ép để làm giảm trọng lượng và tăng tính an toàn của thiết bị. (xem hình bản vẽ lắp)



THIẾT BỊ ĐĨA QUAY

- ① Thân mô tơ
- ② Rô to
- ③ Stator
- ④ Vít giữ mô tơ và tấm đệm
- ⑤ Tấm đệm
- ⑥ Bulông giữ mô tơ với vỏ
- ⑦ Trục gá đĩa
- ⑧ Đĩa mang ảnh đối tượng
- ⑨ Vòng đệm
- ⑩ đai ốc M5
- ⑪ Trục quay mô tơ
- ⑫ Điện trở dây quấn
- ⑬ Núm vận
- ⑭ Thân vỏ

Hình 2. 16: Sơ đồ mạch điện



Hình 2.17: Nguyên lý mạch điện

Sử dụng thiết bị

Lắp đĩa gắn ảnh vật thể

Để thuận tiện, tiết kiệm khi sử dụng, trục gá đĩa của thiết bị đã được thiết kế bằng đường kính trong của các đĩa CD ROM (15mm).



Hình 2.18: Đầu lắp đĩa

Như vậy có thể sử dụng các đĩa CD ROM hỏng dán ảnh của vật thể cần khảo sát lên đĩa ở các vị trí thích hợp. Sau đó đặt vòng đệm và ốc giữ đĩa vặn vào vừa phải. Động cơ được chế tạo có chiều quay ngược chiều kim đồng hồ (không giống chiều quay của các động cơ thông thường). Nên đai ốc mặc dù được vặn vừa phải nhưng càng quay càng ép vào không thể lỏng ra được.



Hình 2.19. Trục mang đĩa quay

Điều chỉnh tốc độ quay

Ngoài khả năng điều chỉnh tốc độ motor bằng cách vặn chiết áp 20Ω người sử dụng còn có thể sử dụng thêm một công tắc phía sau thiết bị để thay đổi tốc độ theo dải. Công tắc này khi bật xuống dưới sẽ nối thêm số vòng vào phần cảm (Stator) của motor làm tốc độ nó giảm đi. Khi này nếu giảm hết chiết áp sẽ có tốc độ đĩa thấp nhất (khoảng 100vg/ph).

Khi công tắc bật lên trên tốc độ motor tăng do số vòng dây phụ bị cắt ra khỏi mạch phần cảm. Lúc này nếu tăng hết chiết áp sẽ có tốc độ cao nhất (3000vg/ph)



Hình 2.20. Nút điều chỉnh phía sau

Bảo quản

- + Bảo quản thiết bị ở nhiệt độ trong phòng
- + Tránh nơi ẩm ướt, tránh nước hoặc chất lỏng chảy vào thiết bị

Tích hợp mô hình thí nghiệm nhận dạng kiểm tra sản phẩm tốc độ cao phục vụ cho nghiên cứu và đào tạo

Hệ thống nhận dạng kiểm tra sản phẩm tốc độ cao cho nghiên cứu và đào tạo đã được chế tạo và tích hợp bao gồm:

- i. Máy PC Pentium IV
- ii. Hệ phần mềm phát triển Instant Vision cho Bi-I
- iii. Bi-I camera kết nối mạng Ethernet
- iv. Thiết bị đĩa quay
- v. Đèn chiếu sáng

Mô hình tích hợp được mô tả trong hình 2.21.



Hình 2.21 Mô hình thí nghiệm nhận dạng kiểm tra sản phẩm tốc độ cao phục vụ cho nghiên cứu và đào tạo

Hệ thống đã được chế tạo tích hợp và lắp đặt trong phòng thí nghiệm chạy ổn định và đã phục vụ cho nhiều nghiên cứu thử nghiệm về CNN và xử lý ảnh nhanh tốc độ cao > 10000fps như trình bày ở các phần sau.

2.2.2. THÍ NGHIỆM PHÂN LOẠI THUỐC VIÊN CHUYỂN ĐỘNG TỐC ĐỘ CAO PILLS

Bài toán

Với mục đích nghiên cứu hệ thống thị giác Bi-I AnaLogic/Eutecus's với chip ACE16k trong bài toán phân loại đối tượng nhỏ đang chuyển động với tốc độ cao, chúng tôi xây dựng mô hình phân loại thuốc viên chuyển động tốc độ cao sử dụng hệ thị giác Bi-IV2 trong phòng thí nghiệm như sau:

Đối tượng (thuốc viên và viên nhộng) có kích thước và hình dáng khác nhau được đặt trong bề mặt của đĩa quay. Đặt Bi-IV2 với mảng chip xử lý cảm biến ở vị trí thích hợp để thu được ảnh rõ nét trong khi đĩa được quay với tốc độ 3000 vòng/phút. Bi-IV2 được kết nối với máy tính PC qua mạng TCP/IP.

Mục đích của thí nghiệm là phân loại mỗi đối tượng dựa trên kích thước và hình dáng của nó và xác định sự chênh lệch giữa hình dáng thực tế và hình dáng chuẩn. Nếu sự chênh lệch đã lớn hơn ngưỡng cần thiết thì chỉ ra rằng 1 phần của thuốc viên hoặc con nhộng đã bị mất. Xây dựng ứng dụng độc lập chạy trên Bi-IV2, thông tin về lỗi được gửi qua mạng Ethernet tới máy tính chủ. Thuật toán nhận biết và phân chia được thực hiện tại mọi thời điểm thuốc viên trong tầm nhìn của cảm biến.

Mô hình trong phòng thí nghiệm

Với bài toán đặt ra, cần tốc độ xử lý ảnh nhanh và ảnh bắt về không cần độ phân giải cao nên sử dụng chip CNN loại 2D ACE16k của Bi-IV2, ảnh thu về có kích thước 128x128. Phần cứng và phần mềm phát triển ứng dụng bao gồm:

- Bi-IV2, Chip ACE16kv2
- Thư viện lập trình Bi-I SDK
- Thư viện xử lý tín hiệu và hình ảnh (SI Lib) của InstantVision ISE
- Thư viện phân loại theo đặc trưng (FCLib)

Hệ phân loại thuốc viên chuyển động tốc độ cao sử dụng công nghệ CNN trong phòng thí nghiệm như trong hình 2.22.



Hình 2.22. Hệ phân loại thuốc viên sử dụng công nghệ CNN trong phòng thí nghiệm

Các viên thuốc với hình dáng, kích thước khác nhau (viên tròn to, tròn nhỏ và con nhộng) được đặt trong một đĩa quay như hình 2.23, vị trí Bi-IV2 với mảng chip xử lý cảm biến thích hợp để thu được ảnh rõ nét với nguồn ánh sáng cung cấp phù hợp, Bi-IV2 kết nối với PC qua mạng Ethernet. Địa chỉ IP của Bi-IV2 là 192.168.0.252.



Hình 2.23. Các viên thuốc trong bề mặt đĩa

Trong đó:

- Bi-IV2 sử dụng ống kính camera có tiêu cự $f = 25\text{mm}$ và đặt cố định và đối diện với đĩa quay.
- Khoảng cách từ đĩa quay tới camera của Bi-IV2 là 36 cm.
- Đĩa quay có thể thay đổi tốc độ khác nhau nhờ chiết áp trên thiết bị mang đĩa quay (từ 40 vòng/phút tới 3000 vòng/phút).
- Các viên thuốc màu trắng (màu sáng), màu nền của đĩa quay là màu đen để làm nổi bật đối tượng.
- Cung cấp nguồn sáng phù hợp cho ứng dụng.

- Máy tính PC có thể kết nối với Bi-IV2 và cài đặt các chương trình hỗ trợ cần thiết (Code Composer Studio v3.1 và gói phần mềm của Bi-I).

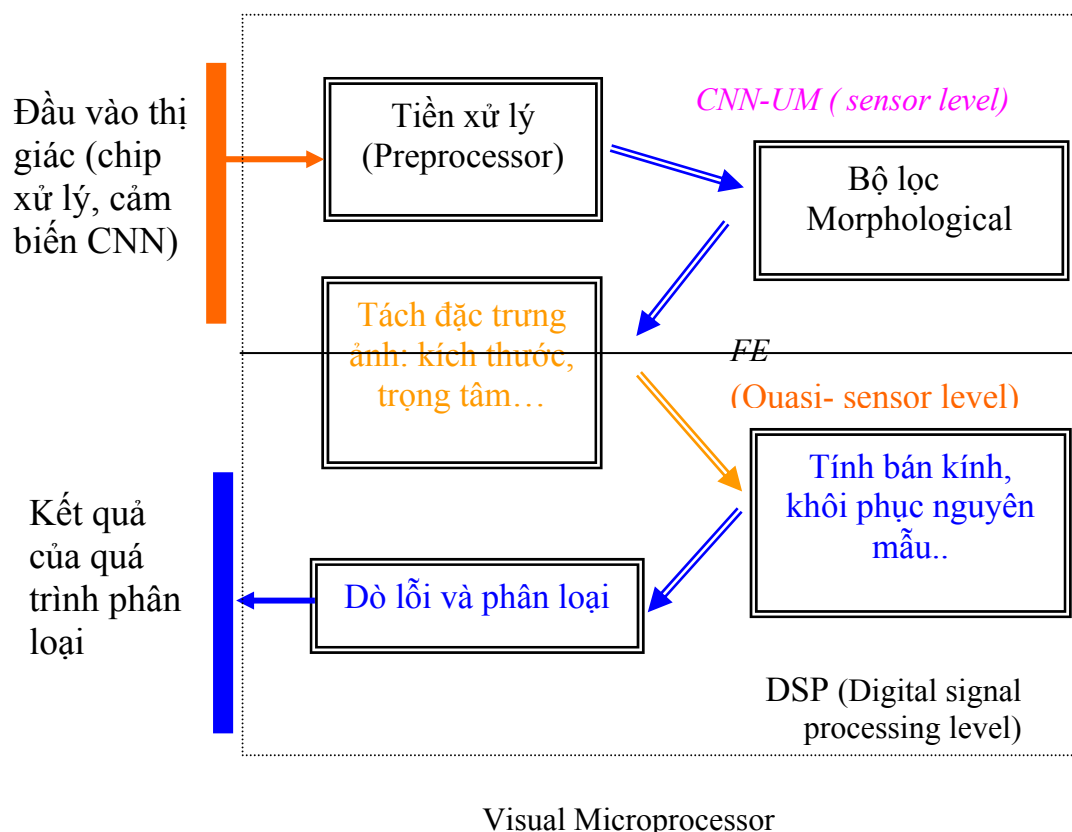
Phát triển phần mềm thí nghiệm phân loại và nhận dạng thuốc viên tốc độ cao PILLS

Môi trường phần mềm phát triển cho ứng dụng là Code Composer Studio v3.1 và các thư viện hỗ trợ xử lý ảnh của Bi-I: thư viện lập trình Bi-I SDK, thư viện xử lý tín hiệu và hình ảnh (SI Lib), thư viện phân loại theo đặc trưng (FCLib) của InstantVision ISE.

Cài đặt chương trình Code Composer Studio v3.1 trên máy tính PC. Trong cửa sổ Code Composer Studio Setup, mục *My System*, chọn họ *C6415 Device Cycle Accurate Simulator*, lưu và thoát khỏi mục này, khởi động chương trình Code Composer Studio và tiến hành tạo lập project với tên PillDemo.pjt phát triển cho ứng dụng với giải thuật trình bày sau đây.

Thuật toán

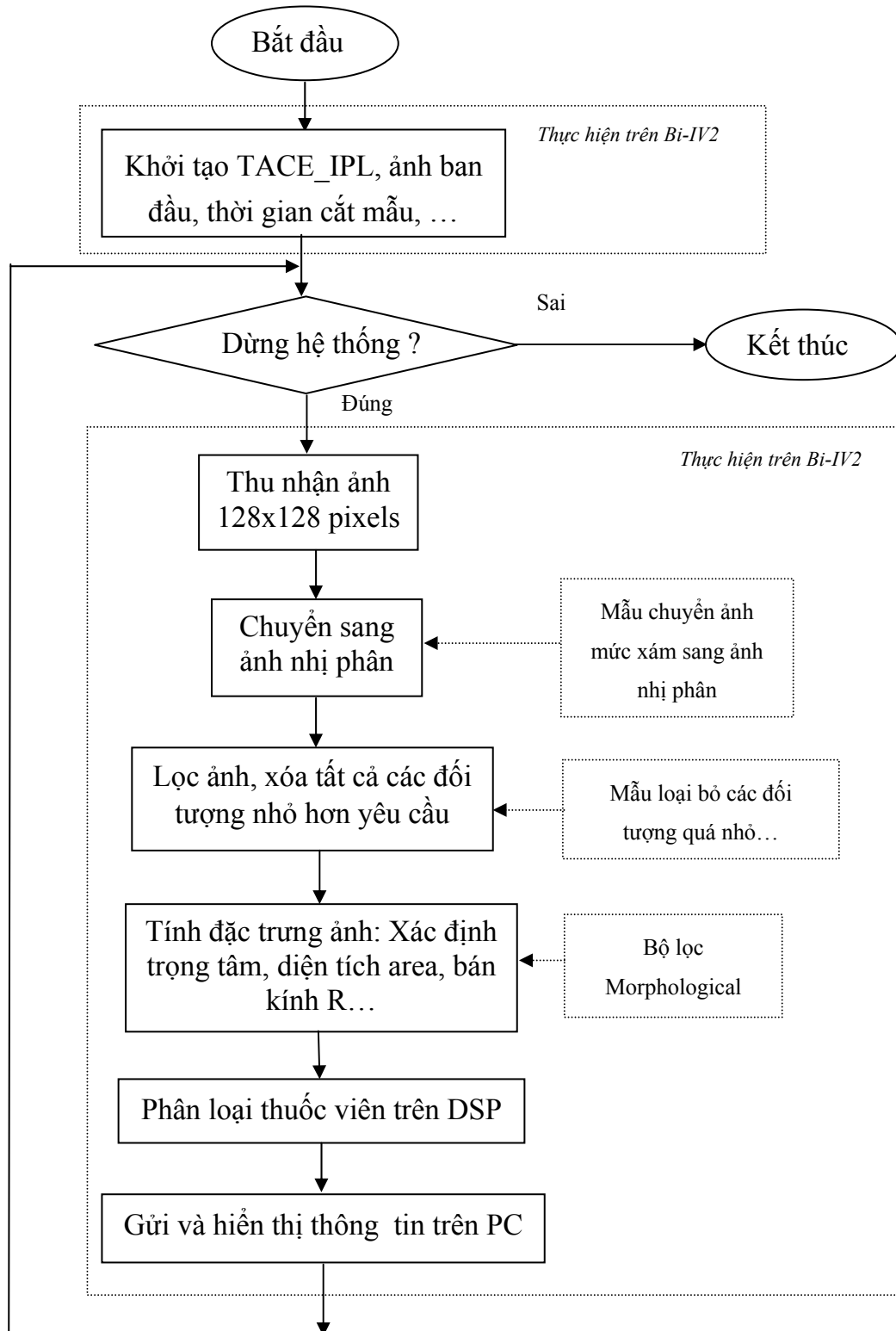
Sơ đồ các bước xử lý trên Bi-IV2 như hình 2.24:



Hình 2.24. Các bước xử lý trên Bi-IV2

Với bài toán đã đặt ra, các phép xử lý ảnh và phân loại đều được thực hiện trên Bi-I V2, vì thế sau khi lấy ảnh mức xám (128x128 pixels) từ chip thị giác, bước tiền xử lý tiến hành chuyển ảnh mức xám này sang ảnh nhị phân, sau đó thực hiện phép lọc ảnh: xóa bỏ các đối tượng nhỏ hơn yêu cầu... để làm chất lượng ảnh tốt hơn. Qua bộ lọc hình thái học, tính toán được các thông tin về hình dáng, kích thước, vị trí. Sự phân loại được dựa trên các đặc trưng như hình dáng, kích thước... của đối tượng. Với các viên thuốc tròn, từ ảnh thu được có thể dễ dàng xác định được bán kính hình tròn nguyên mẫu của các viên thuốc (khôi phục hình dạng

nguyên mẫu), từ đó có thể xác định được phần thiếu hụt của đối tượng cho các viên tròn bị vỡ. Giữa các bước này, chỉ quá trình khôi phục mẫu và sự phân chia cuối cùng được thực hiện bởi DSP, các bước tính toán còn lại được xử lý song song trong chip ACE16k.



Hình 2.25. Lưu đồ chương trình ứng dụng phân loại thuốc viên chuyển động tốc độ cao sử dụng Bi-IV2

Như ta đã biết, có thể đặt trực tiếp trạng thái chạy/dừng trong Bi-I SDK từ ứng dụng chủ (trong phần mềm IVRun, nút Run/Stop đặt trạng thái chạy/ dừng trong Bi-IV2). Chạy ứng dụng với phần mềm IVRun, chương trình sẽ tự nạp xuống Bi-IV2 qua mạng Ethernet. Đầu tiên, chương trình sẽ khởi tạo các thông số cần thiết như ảnh ban đầu, thời gian cắt mẫu, thư viện xử lý ảnh TACE_IPL,...Thực hiện hàm GetStopStatus() để nhận lại trạng thái dừng của phần mềm IVRun. Nếu giá trị nhận về bằng 0 (trạng thái Run), chương trình sẽ bắt ảnh, thực hiện các thuật toán phân loại thuốc viên trên Bi-IV2 và quá trình này tiếp tục lặp lại. Thông tin của quá trình luôn được cập nhật gửi về PC qua mạng Ethernet sau mỗi lần bắt ảnh và phân loại. Nếu GetStopStatus() nhận về giá trị bằng 1 (trạng thái Stop), phần mềm IVRun ra lệnh cho chương trình trên Bi-IV2 ngừng hoạt động.

Lưu đồ chương trình ứng dụng phân loại thuốc viên chuyển động tốc độ cao sử dụng Bi-V2 như trong hình 2.25.

Trong ứng xử lý ảnh nhanh này, sử dụng chip CNN loại 2D ACE16K của Bi-IV2, ảnh mức xám thu về từ chip thị giác có kích thước 128x128. Với mỗi giải thuật xử lý ảnh sử dụng trong ứng dụng, điều quan trọng là cần thiết lập các thông số của mẫu (A, B, z) phù hợp cho ACE16k, ở đây A, B đều là những ma trận kích thước 3x3 có dạng như sau:

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix} \quad B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}$$

Sau đây là các thuật xử lý ảnh và các mẫu (A, B, z) tương ứng của nó đã sử dụng trong chương trình ứng dụng.

Tiền xử lý

Tiền xử lý chính là bước xử lý ảnh đầu tiên, ảnh thu về chuyển sang ảnh nhị phân và lọc ảnh, xóa bỏ các đối tượng có kích thước quá nhỏ.

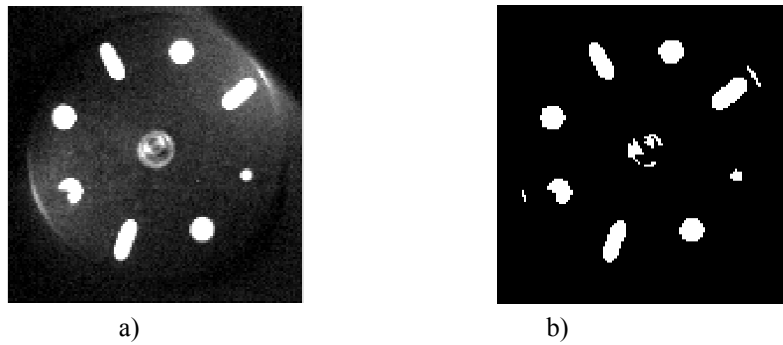
➤ Chuyển ảnh mức xám sang ảnh nhị phân

Mẫu chuyển ảnh mức xám sang ảnh nhị phân như sau:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -z^* \quad (-1 < z^* < 1)$$

- Đặc điểm: - Dựa vào ảnh mức xám P và ngưỡng z^* .
 - Đầu vào: U(t) tùy ý hoặc mặc định U(t)=0.
 - Trạng thái ban đầu: X(0)=P.
 - Đầu ra: Y(t) => Y(∞) = ảnh nhị phân khi tất cả các điểm ảnh P mức xám có $p_{ij} > z^*$ trở thành màu đen.

Ảnh mức xám đầu vào và ảnh nhị phân sau khi qua ngưỡng như ở hình 2.26.



Hình 2.26. Ảnh mức xám và ảnh sau khi qua ngưỡng

➤ Xóa các đối tượng nhỏ hơn yêu cầu

Tác vụ này sẽ xóa đi các điểm ảnh mà tổng số điểm ảnh lân cận nó nhỏ hơn hoặc bằng

4. Mẫu xóa các đối tượng nhỏ hơn yêu cầu như sau:

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0.$$

- Đặc điểm: - ảnh nhị phân P
 - Đầu vào $U(t) = P$.
 - Trạng thái ban đầu $X(0) = P$.
 - Điều kiện đường biên: loại cố định $y_{ij} = 0$ cho tất cả các tế bào ảo, biểu thị bởi $Y[0] = 0$.
 - Đầu ra $Y(t) \Rightarrow Y(\infty) =$ ảnh nhị phân P đã bỏ các đối tượng nhỏ.

Ảnh nhị phân trước và sau khi xóa các đối tượng nhỏ như trong hình 2.27.



Hình 2.27. Ảnh nhị phân trước và sau khi xóa các đối tượng nhỏ

Tác vụ hình thái học

Hoạt động cơ bản của hình thái học ảnh nhị phân chính là phép ăn mòn (erosion) và làm đầy (dilation).

➤ Phép ăn mòn

Mẫu (A, B, z) cho mạng CNN trong Bi-I thực hiện phép ăn mòn như sau:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix} \quad z = z_E.$$

trong đó: $b_{k,l} = 0$ hoặc 1 (không được tất cả là 0).

$$z_E = 0.5 - p_1, \quad p_1 \text{ là tổng những giá trị 1 trong mẫu } B, \quad p_1 > 0.$$

- Đặc điểm: Ảnh nhị phân đầu vào P và phần tử cấu trúc ζ_E . ζ_E là mẫu 3x3 đen và trắng. Toàn bộ các điểm đen trong ζ_E trùng khớp với các điểm đen trong mẫu B (đen = 1). ζ_E cũng được gọi là mẫu cấu trúc.

- Đầu vào: $U(t) = P$.

- Trạng thái ban đầu: $X(0) = 0$.

- Điều kiện đường biên: $y_{kl} = -1, u_{kl} = -1$ cho tất cả các tế bào $C(k,l)$ đường biên.

- Đầu ra: $Y(t) \Rightarrow Y(\infty) =$ ảnh nhị phân mà mỗi điểm ảnh $(i,j) =$ đen nếu mẫu cấu trúc đã chuyển $\zeta_E(i + \alpha, j + \beta)$ vừa vận trong P sau khi chuyển ζ_E sang điểm ảnh (i, j) với trọng tâm của ζ_E giữ tại $C(i, j)$ (ví dụ $\zeta_E(x + \alpha, y + \beta) \in P$), ở đây, α, β là tọa độ cục bộ liên quan của ζ_E .

➤ Phép làm đầy

Mẫu (A, B, z) cho phép làm đầy như sau:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix} \quad z = z_D.$$

trong đó: $b_{k,l} = 0$ hoặc 1.

$$z_D = p_1 - 0.5, \quad p_1 \text{ là tổng những giá trị 1 trong mẫu } B, \quad p_1 > 0.$$

- Đặc điểm
 - Ảnh nhị phân P và mẫu 3x3 đen và trắng. Mẫu này gọi là mẫu cấu trúc ζ_D .
 - Đầu vào: $U(t) = P$.
 - Trạng thái ban đầu: $X(0) = 0$
 - Điều kiện biên: $y_{kl} = -1, u_{kl} = -1$ cho tất cả các tế bào $C(k,l)$ đường biên.
 - Đầu ra: $Y(t) \Rightarrow Y(\infty) =$ ảnh nhị phân tạo ra như sau:

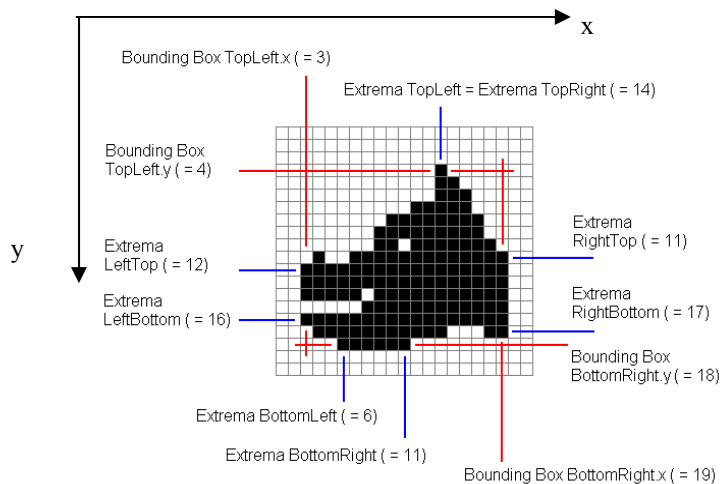
Bước 1: Mỗi điểm ảnh màu đen $C_b(i,j) \in P$ tại tọa độ (i,j) chuyển sang mẫu cấu trúc ζ_D và kết quả được tập đã chuyển các điểm ảnh màu đen $\zeta_D(i, j)$.

Bước 2: Đặt tập $\zeta_D(i, j)$ với mọi (ij) mà $u_{ij} = 1$ trên tất cả các điểm ảnh đen $C_b(i,j) \in P$ sẽ được đầu ra $y(\infty)$.

$$Y(\infty) = \bigcup_{\text{all}(ij), u_{ij}=1} \zeta_D(ij)$$

Tính đặc trưng đối tượng ảnh

Các đặc trưng có thể sắp xếp thành 2 nhóm chính. Những đặc trưng trong nhóm thứ nhất được tính toán trực tiếp từ đối tượng, trong khi nhóm thứ hai bao gồm một vài thuộc tính của hình elíp có cùng mô men thứ hai với đối tượng. Hình 2.28 cho thấy những đặc trưng được đo trực tiếp từ đối tượng.



Hình 2.28. Những đặc trưng được đo trực tiếp từ đối tượng

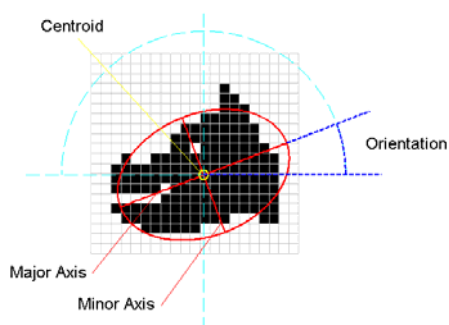
Các đặc trưng được tính toán trực tiếp từ đối tượng như sau:

Diện tích	Là diện tích đối tượng, bằng số pixel thuộc về đối tượng.
Đường kính tương đương	Là đường kính của vòng tròn có chung diện tích với đối tượng.
Hộp bao quanh	Là các tọa độ của góc trên cùng bên trái và góc dưới cùng bên phải của hình chữ nhật nhỏ nhất (với các cạnh song song với mép của ảnh) bao quanh đối tượng.
Phạm vi	Là đơn vị đo xem đối tượng lấp đầy khoảng không nó sử dụng như thế nào. Là thương số của diện tích đối tượng và diện tích của hộp bao quanh.
Cực trị	Là những điểm cực của đối tượng trên hộp bao quanh.

Những đặc trưng đo từ hình elíp có cùng moment thứ hai với đối tượng:

Trọng tâm	Là trung tâm của elíp, trùng với tâm hình học của đối tượng
Chiều dài trục lớn	Là chiều dài trục lớn của elíp.
Chiều dài trục nhỏ	Là chiều dài trục nhỏ của elíp.
Độ lệch tâm	Là tỉ lệ của khoảng cách tập trung và chiều dài trục chính của elíp. Đường tròn có độ lệch tâm là 0, đường thẳng là 1.
Hướng	Là góc của trục lớn tương ứng với chiều dương trục hoành.

Hình 2.29 cho thấy các đặc trưng được đo từ hình elíp có cùng mô men thứ hai với đối tượng:



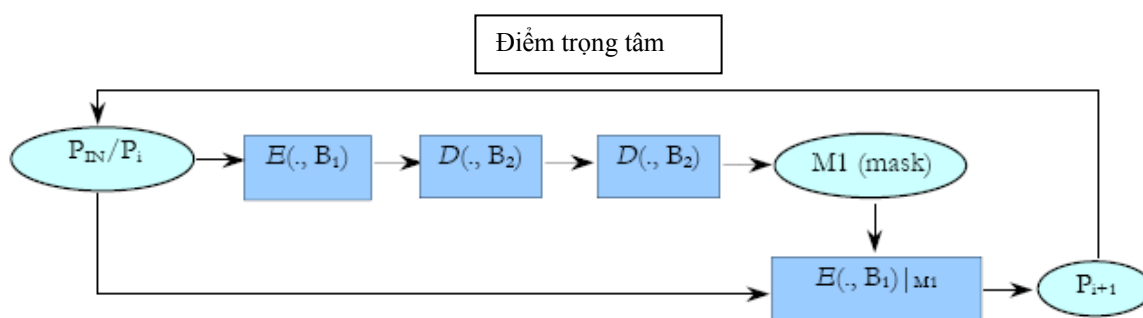
Hình 2.29. Những đặc trưng được đo từ hình elíp có cùng moment thứ hai với đối tượng

Vì các viên thuốc có dạng tròn hoặc nhộng, ta chọn các đặc trưng ảnh như: vị trí tâm, tổng điểm ảnh, chu vi bao quanh... dựa trên ảnh sau khi qua bước tiền xử lý.

➤ Tìm điểm trong tâm

Phương pháp tìm điểm trọng tâm như sau: Tác vụ hình thái học tạo ra mặt nạ ăn mòn (masked erosion), hình ảnh chứa đối tượng có kích thước khác nhau sẽ có số bước hình thái học khác nhau. Sử dụng phép ăn mòn thông thường, các đối tượng nhỏ hơn sẽ không xuất hiện trước khi tìm thấy các điểm trọng tâm của chúng trong khi các đối tượng lớn hơn vẫn chứa nhiều điểm ảnh. Mặt nạ ăn mòn có thể ngăn chặn các đối tượng nhỏ không xuất hiện: tập các đối tượng này sẽ triệt tiêu với ứng dụng tiếp sau của toán tử ăn mòn.

Một bước của mặt nạ ăn mòn có thể sử dụng nhiều lần. Phương pháp tìm điểm trọng tâm sử dụng tác vụ hình thái học như trong hình 2.30 với B_1 và B_2 biểu thị cho các phần tử cấu trúc sử dụng trong phép ăn mòn $E(.)$ và làm dày $D(.)$.



Hình 2.30. Tìm điểm trọng tâm của đối tượng sử dụng tác vụ hình thái học.

Mặt nạ M1 là ảnh đen-trắng trong đó các điểm ảnh đen chỉ các vị trí mà tác vụ hình thái học có hiệu quả. Các điểm ảnh tại các vị trí điểm ảnh mặt nạ màu trắng không thay đổi. Ở đây, mặt nạ chứa các đối tượng lớn, toàn bộ chúng sẽ không hiển thị trong suốt quá trình ăn mòn. Tạo mặt nạ trong 3 bước, đầu tiên, toán tử ăn mòn sẽ xóa các đối tượng nhỏ hơn, sau đó

2 tác vụ làm dày tiếp theo tạo ra các đối tượng lớn hơn kích thước gốc của chúng. Ảnh mặt nạ M1 này cho phép ăn mòn chỉ ở các vị trí của các đối tượng lớn hơn.

Mẫu thực hiện CNN của phép ăn mòn là:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0.5 & 1 & 0.5 \\ 1 & 1 & 1 \\ 0.5 & 1 & 0.5 \end{bmatrix} \quad z = -6.5$$

Mẫu thực hiện CNN của phép làm dày là:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad z = 8.5$$

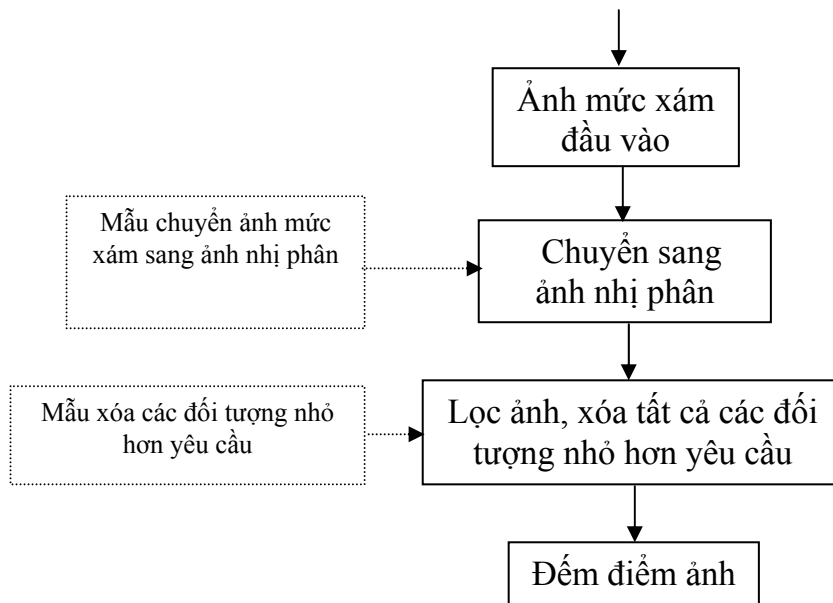
Một điều rất quan trọng là phép làm dày cần thực hiện ít nhất 2 lần. Hình 2.31 cho thấy ảnh nhị phân đầu vào và ảnh vị trí các điểm trọng tâm của các đối tượng sử dụng phép hình thái học.



Hình 2.31. Ảnh nhị phân đầu vào và ảnh vị trí các điểm trọng tâm của các đối tượng

Tính diện tích đối tượng ảnh




Để tính diện tích của đối tượng ảnh (bằng tổng số điểm ảnh), ảnh mức xám đầu vào qua phép chuyển đổi sang ảnh nhị phân. Sử dụng phép lọc để xóa nhiễu, sau đó thực hiện phép đếm điểm ảnh. Lưu đồ tính diện tích của đối tượng ảnh như hình 2.32.



Hình 2.32. Lưu đồ xác định diện tích của đối tượng ảnh

Phân loại

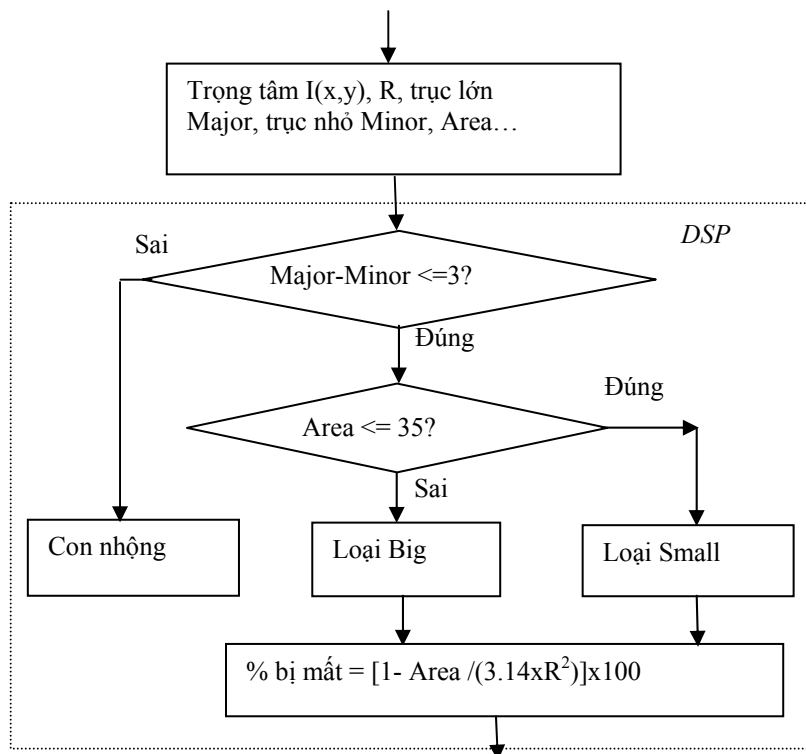
Để phân loại có thể dựa vào các đặc trưng của các đối tượng. Từ các đặc tính của đối tượng được tính ở phần trên như tọa độ trọng tâm $I(x,y)$, trục lớn Major, trục nhỏ Minor, tổng số điểm ảnh trên đối tượng Area ... quá trình phân loại và tính toán phân bị vỡ sẽ dễ dàng thực hiện trên DSP. Từ sự chênh lệch về kích thước (số điểm ảnh của đối tượng hay diện tích của đối tượng trên ảnh thu được) có thể phân loại dạng loại viên to/nhỏ. Với các viên tròn tỷ lệ Minor/Major xấp xỉ 1 vì vậy từ độ lệch về hình dáng (Major-minor) có thể phân biệt viên dạng tròn hay dạng nhộng. Các cách phân loại như sau:

- Tìm độ lệch về kích thước: 
- Tìm độ lệch về hình dáng: 
- Tìm độ lệch về bề mặt: 

Đối với viên hình tròn, phần trăm phân bị mất tính như sau:

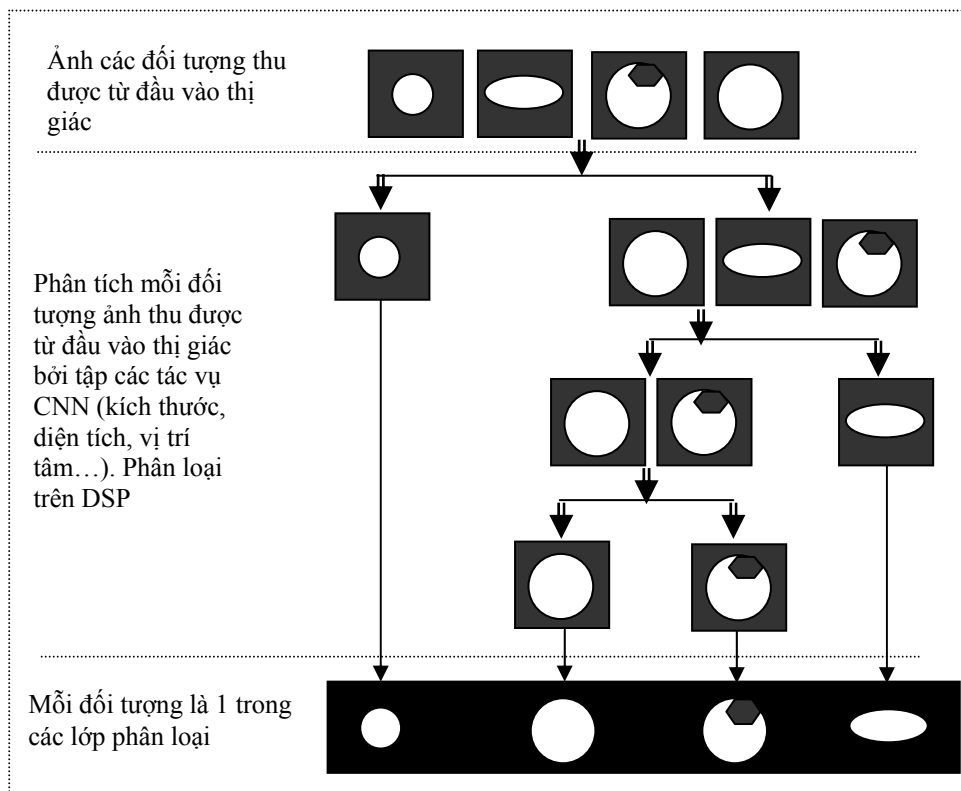
$$\% \text{ bị mất} = [1 - \text{Area} / (3.14 \times R^2)] \times 100$$

Trong đó Area là diện tích thực sự viên thuộc có trên ảnh thu được, R là bán kính. Lưu đồ thuật toán xử lý trên DSP như hình 2.33:



Hình 2.33. Lưu đồ chương trình phân loại trên DSP

Quá trình phân loại thuốc viên có thể minh họa như hình 2.34.

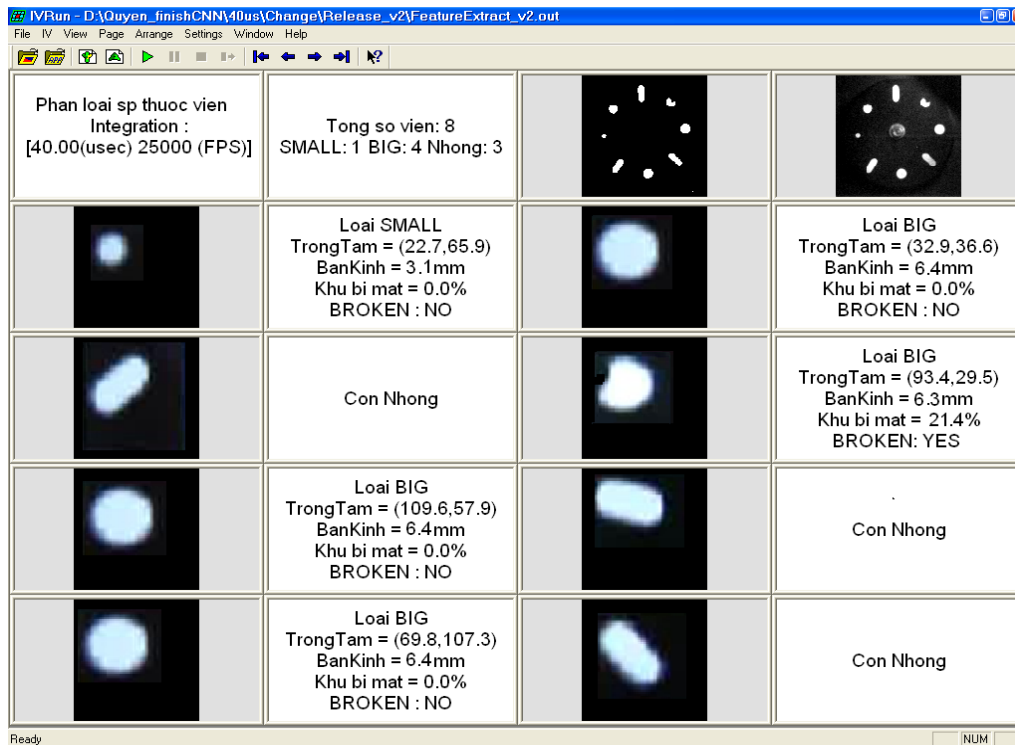


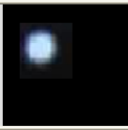

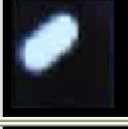
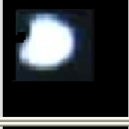
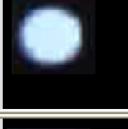

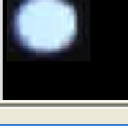
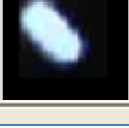
Hình 2.34. Hình ảnh minh họa cho quá trình phân loại thuốc viên

KẾT QUẢ ĐẠT ĐƯỢC TRONG PHÒNG THÍ NGHIỆM

Với ánh sáng cung cấp như trong điều kiện phòng thí nghiệm, Bi-IV2 bắt ảnh với tốc độ 25 000 fps. Thời gian bắt một ảnh là $40\mu\text{s}$, tổng thời gian xử lý trung bình cho quá trình phân loại là 17,1 ms với 8 đối tượng ảnh.

Kết quả đạt được trên cửa sổ giao diện của PC như hình 2.35. Giao diện hiển thị tốc độ bắt giữ ảnh của Bi-IV2, số lượng viên thuốc của từng loại, tổng số viên đếm được trên đĩa quay và ảnh kèm các thông tin về từng viên thuốc. Các thông tin luôn được cập nhật sau mỗi lần bắt ảnh và phân loại.



Phan loai sp thuốc viên Integration : [40.00(usec) 25000 (FPS)]		Tổng số viên: 8 SMALL: 1 BIG: 4 Nhung: 3	
	Loại SMALL TrongTam = (22.7,65.9) BanKinh = 3.1mm Khu bị mat = 0.0% BROKEN : NO		Loại BIG TrongTam = (32.9,36.6) BanKinh = 6.4mm Khu bị mat = 0.0% BROKEN : NO
	Con Nhung		Loại BIG TrongTam = (93.4,29.5) BanKinh = 6.3mm Khu bị mat = 21.4% BROKEN: YES
	Loại BIG TrongTam = (109.6,57.9) BanKinh = 6.4mm Khu bị mat = 0.0% BROKEN : NO		Con Nhung
	Loại BIG TrongTam = (69.8,107.3) BanKinh = 6.4mm Khu bị mat = 0.0% BROKEN : NO		Con Nhung

Hình 2.35. Kết quả thu được trong bài toán phân loại thuốc viên

Thông tin về một viên thuốc hiển thị trên máy tính PC như sau:

- Loại thuốc (dạng to, hay nhỏ hay con nhộng).
- Vị trí tâm của viên thuốc.
- Bán kính của viên thuốc.
- Phần bị mất (tính theo %).
- Đánh giá: nguyên vẹn hay bị vỡ mất góc.

Ảnh hiển thị từng viên thuốc tương ứng ở các thời điểm là ảnh mức xám được sao chép sang ảnh (64x64).

Như vậy, với bài toán phân loại thuốc viên chuyển động tốc độ cao, tốc độ của đĩa quay (3000 vòng/phút), nếu hệ thống sử dụng camera thông thường (24 ảnh/s) thì không thể đáp ứng được, sử dụng công nghệ CNN ở đây là phù hợp, với cường độ ánh sáng đủ mạnh, hệ thống hoạt động tốt, thông tin về những viên thuốc là khá đầy đủ, xác định viên thuốc nào phù hợp hay không dựa trên việc phân tích hình dáng, kích thước của viên thuốc.

2.2.3 HƯỚNG DẪN SỬ DỤNG THÍ NGHIỆM PHÂN LOẠI THUỐC VIÊN TỐC ĐỘ CAO PILLS

Phần mềm thử nghiệm phân loại thuốc viên tốc độ cao PILLS đã được phát triển và nạp trong đĩa **CD “ các mô hình và chương trình thử nghiệm công nghệ xử lý ảnh tốc độ cao phục vụ cho nghiên cứu và đào tạo ”** của nhiệm vụ nghị định thư.

Mô hình thí nghiệm ứng dụng công nghệ mạng nơ ron tế bào cho phân loại thuốc viên chuyển động tốc độ cao có thể nhận biết và phân loại thuốc viên (đặt rời nhau đang chuyển động tốc độ cao: ≤ 3000 vòng/phút trên 1 đĩa quay) dựa trên kích thước và hình dáng của nó và xác định sự chênh lệch giữa hình dáng thực tế và hình dáng chuẩn. Nếu sự chênh lệch đã lớn hơn ngưỡng cần thiết thì chỉ ra rằng một phần của thuốc viên hoặc con nhộng đã bị mất. Thông tin về lỗi được gửi qua mạng Ethernet tới máy tính chủ PC. Thuật toán nhận biết và phân chia được thực hiện tại mọi thời điểm thuốc viên trong tầm nhìn của cảm biến.

- Ứng dụng công nghệ mạng nơ ron tế bào sử dụng hệ thị giác Bi-iV2 trong phân loại thuốc viên đã đảm bảo được tốc độ xử lý nhanh mà các camera thường không đáp ứng được.

- Chương trình cũng có thể dễ dàng thích nghi với bài toán phân chia và kiểm tra chất lượng với những đối tượng nhỏ khác...

Yêu cầu cài đặt và thử nghiệm

Để thử nghiệm mô hình ứng dụng công nghệ mạng nơ ron tế bào cho phân loại thuốc viên chuyển động tốc độ cao hoạt động cần:

- Đĩa CD cài đặt bao gồm gói phần mềm của Bi-i và chương trình ứng dụng phân loại thuốc viên.

- Một máy tính PC với hệ điều hành Windows 2000 hoặc XP, một số phần có thể yêu cầu Linux.

- Hệ thị giác Bi-iV2 với chip ACE 16k kết nối với PC qua mạng Ethernet.

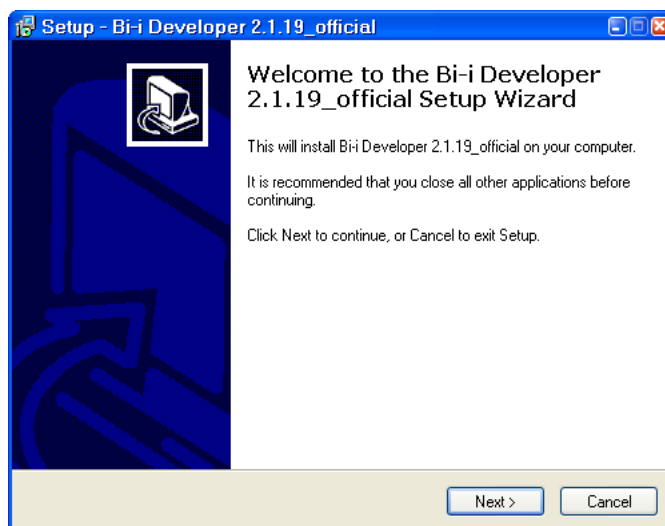
- Một thiết bị thử nghiệm với các thuốc viên đặt rời nhau trên 1 đĩa quay (tốc độ ≤ 3000 vòng/phút)

- Cung cấp ánh sáng cho hệ thống.

Gói phần mềm Bi-i

Gói phần mềm Bi-i có 2 phiên bản: introductory và developer. Gói introductory có phần mềm để viết, dịch và chạy chương trình AMC và cùng có bao gồm nhiều thư viện và phiên bản thử nghiệm của Bi-i SDK. Gói developer cũng hỗ trợ việc lập trình API và Bi-i SDK. Gói phần mềm Bi-i cần Microsoft Windows 2000 hoặc XP.

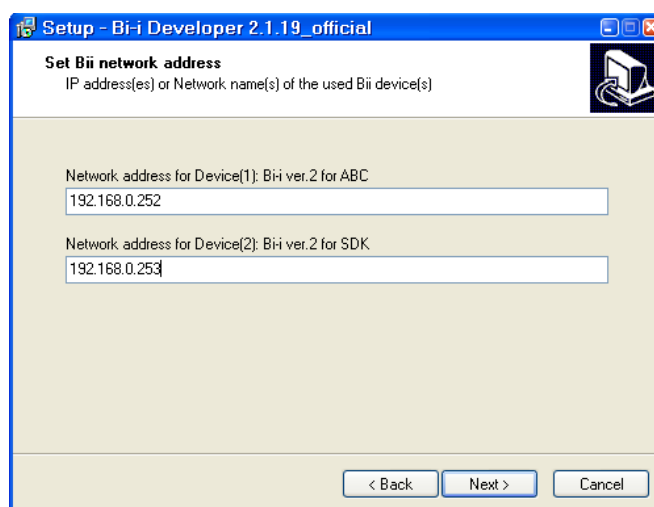
Trước tiên, cần cài đặt gói phần mềm trên máy tính PC kết nối với Bi-i qua mạng Ethernet. Giao diện khi cài đặt gói phần mềm như sau:



Hình 2.36. Cửa sổ khi cài đặt gói phần mềm Bi-i

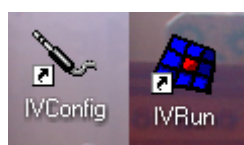
Trong quá trình cài đặt, ta cần nhập địa chỉ IP của Bi-i tùy theo các ứng dụng chạy trên nó. Ở đây, Bi-i có địa chỉ 192.168.0.252 cho ứng dụng ABC, Bi-i có địa chỉ 192.168.0.253 cho ứng dụng SDK. Các đặc tính của Bi-i có thể thay đổi lại nhờ IVConfig.

Cửa sổ đặt địa chỉ mạng của Bi-i với các ứng dụng ABC hay SDK như trong hình sau:



Hình 2 37. Cửa sổ đặt địa chỉ mạng của Bi-i với các ứng dụng ABC hay SDK

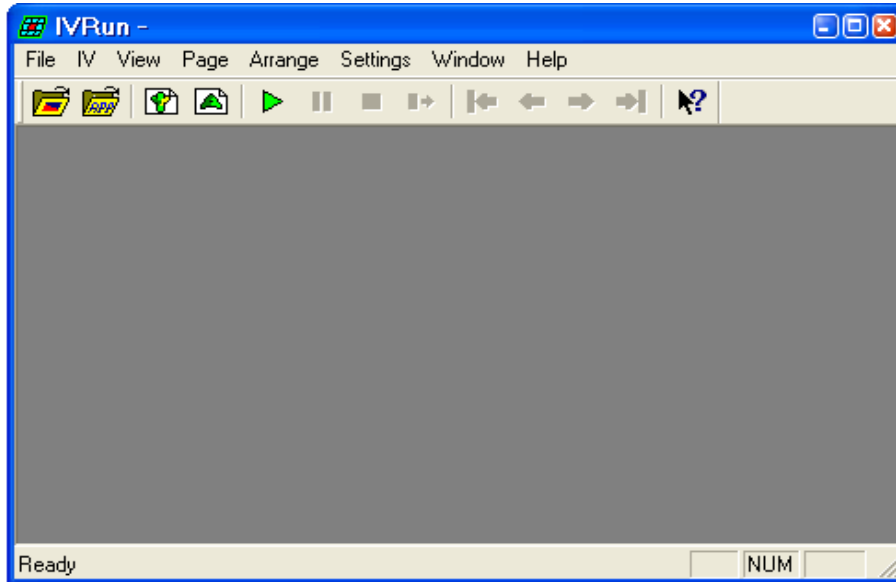
Sau khi cài đặt gói phần mềm Bi-i, ta cần tìm hiểu thêm về IVRun và IVConfig. Biểu tượng của chúng trên desktop của máy tính như sau:



Cũng có thể tìm thấy IVConfig qua Start/Control Panel của máy tính.

IVRun

IVRun là môi trường đồ họa cho các ứng dụng cơ bản chạy trên InstantVision. IVRun hiển thị hình ảnh và các lời thoại dạng văn bản trong các cửa sổ riêng biệt. Cửa sổ chính của chương trình IVRun như hình 2.38.



Hình 2.38. Cửa sổ chính của chương trình IVRun

Chạy ứng dụng

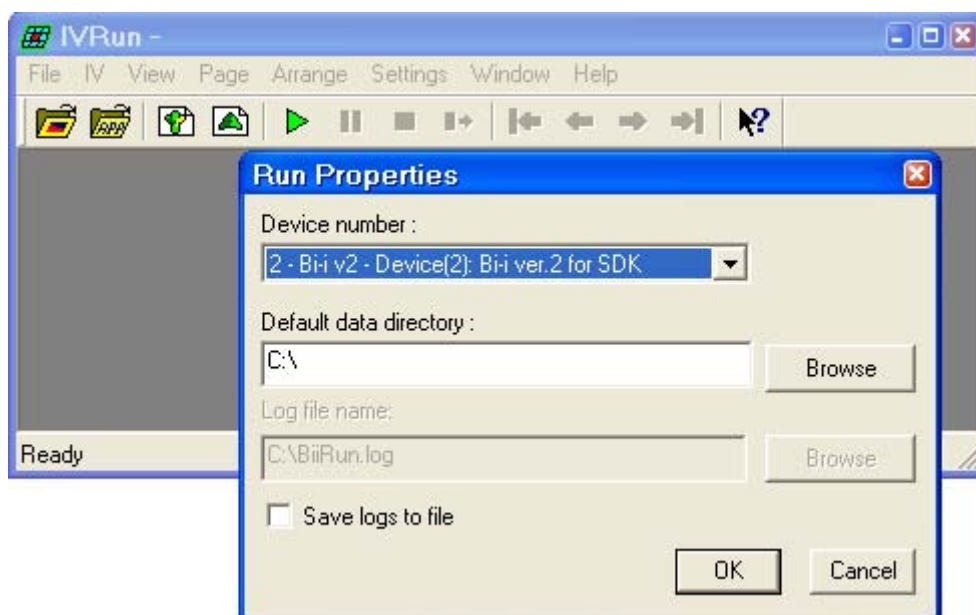
Để chạy ứng dụng trên IVRun, cần các điều kiện sau:

- Thiết bị Bi-i cần đặt cấu hình trong hệ thống với IVConfig hoặc BiiConfig.
- Số hiệu thiết bị (được đặt bởi IV/BiiConfig) được lựa chọn trong IVRun, trong hộp thoại Run Properties (danh mục Settings, mục Run).
- Nếu ứng dụng mặc định không đặt trong thiết bị thì ứng dụng cần được chỉ rõ (danh mục File, mục Open Application).

Số hiệu thiết bị và tệp ứng dụng cũng được đặt trong dòng lệnh, hoặc chúng có thể là một phần của workspace.

- **Hộp thoại Run Properties**

Để lựa chọn thiết bị có số hiệu tương ứng để chạy ứng dụng cần gọi hộp thoại Run Properties qua mục Run của Settings. Hộp thoại xuất hiện như trong hình 2.39:



Hình 2.39. Hộp thoại Run Properties

Trong đó:

<i>Device number</i>	Số hiệu thiết bị được sử dụng cho thực hiện chương trình (nó được đặt trong IV/BiiConfig). Trong hộp chọn combo, có danh sách các thiết bị đã được cấu hình theo dạng số -loại – chú thích.
<i>Default data directory</i>	Thư mục mặc định cho dữ liệu máy chủ. Tất cả các hoạt động trên máy chủ, các tệp đều được lưu trong thư mục này, ngoại trừ các tệp có đường dẫn đầy đủ.
<i>Log file name</i>	Tên của tệp mà tất cả các chú giải được lưu lại đó nếu đặc tính này được cho phép.
<i>Save logs to file</i>	Cho phép lưu các chú giải vào trong tệp.

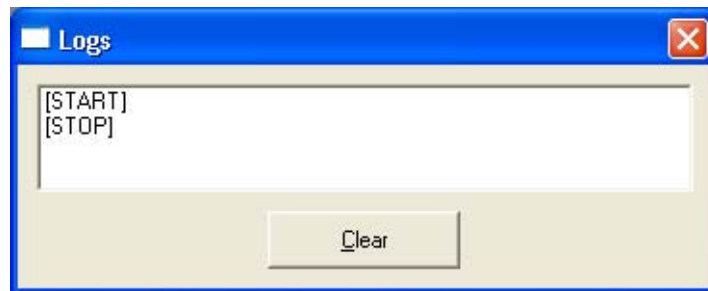
- **Run controls**

Chọn mục Run từ thực đơn Bi-I để chạy ứng dụng trên thiết bị đã chọn. Cho phép dừng, tạm dừng hay bắt đầu lại với chức năng thích hợp của các mục trong IV. Ứng dụng có thể đợi trạng thái, có thể kết thúc với mục WaitEnd của danh mục IV (hoặc nó có thể tự kết thúc nếu như đã vượt qua ngưỡng timeout). Thanh công cụ có các chức năng Run, Pause/Resume, Stop và WaitEnd, sử dụng nó thay vì phải lựa chọn các chức năng trong danh mục của IV.

Nếu ứng dụng mặc định không được đặt trong thiết bị thì ứng dụng phải được chỉ rõ (danh mục File, mục Open Application) hoặc mục Run của danh mục IV vẫn không cho phép kích hoạt. Tên của tệp ứng dụng được hiển thị trên header của cửa sổ chính.

- **Cửa sổ Log**

Cửa sổ Log chứa các thông tin dạng văn bản xuất hiện trong suốt quá trình thực hiện ứng dụng. Sử dụng chức năng Log của danh mục View để hiển thị hay ẩn cửa sổ Log. Cửa sổ Log có thể đặt ở bất kỳ đâu, ngay cả khi ở phía ngoài màn hình chính của IVRun. Cửa sổ Log không có chế độ hiển thị ma trận. Tất cả các thông tin trên cửa sổ bị xoá khi ấn nút Clear trên cửa sổ, tuy nhiên cửa sổ Log cũng tự động xoá khi ứng dụng mới bắt đầu. Các thông điệp Log có thể được lưu lại vào tệp nếu trên hộp thoại Run Properties cho phép chức năng “Save logs to file”. Cửa sổ Log như trong hình 2.40.



Hình 2.40. Cửa sổ Log

Làm việc với Windows

Cửa sổ hình ảnh và dạng văn bản có thể được tạo ra tự động khi ứng dụng đang chạy với đặc tính đặt mặc định. Người sử dụng có thể tạo ra cửa sổ và đặt lại các đặc tính của nó như kích thước, vị trí, màu sắc... và sắp xếp các cửa sổ theo ý thích riêng. Lúc đó, hình ảnh, thông tin dạng text gửi đến khi chạy ứng dụng sẽ được hiển thị trong các cửa sổ này và lưu trữ hình ảnh nếu muốn.

Các ứng dụng có thể chạy mà không cần tạo cửa sổ từ trước, tuy nhiên có thể tùy chọn trên giao diện của IVRun với những cửa sổ đã được tạo ra từ trước. Mỗi cửa sổ có thể nhận ra nhờ số hiệu trên góc trái của cửa sổ. Số hiệu này phải được sử dụng khi ứng dụng gửi dữ liệu tới IVRun.

Cấu hình và workspaces

- **Cấu hình**

Cấu hình lưu trữ trong IVRun để chắc chắn rằng sự thiết lập cuối cùng luôn được gọi khi IVRun khởi động. Cấu hình hoạt động nếu không sử dụng workspaces. Cấu hình được lưu trữ trong tệp IVRun.cfg (trong cùng thư mục chứa IVRun.exe). Cấu hình bao gồm:

- Kích thước và loại của cửa sổ chương trình.
- Trạng thái của thanh công cụ.
- Trạng thái của thanh trạng thái.
- Trạng thái của cửa sổ Log.
- Đặc tính hoạt động.
- Đặc tính hiển thị.

- Đặc tính hình ảnh mặc định.
- Đặc tính văn bản mặc định.

- **Workspaces**

Workspaces trong IVRun được sử dụng cho việc lưu trữ trạng thái của môi trường IVRun. Tập Workspaces chứa các thông tin sau:

- Tất cả các thiết lập được lưu trong cấu hình.
- Vị trí, kích thước và thiết lập của cửa sổ hình ảnh và văn bản.
- Nội dung của các cửa sổ hình ảnh và văn bản nếu nó được tạo ra với câu lệnh New Text hoặc New Image.
- Đường dẫn của tập workspaces.

Tên của workspaces hiện tại được hiển thị trong tiêu đề của cửa sổ chính.

Lựa chọn chức năng Open Workspace từ File để mở workspace. Đọc tập workspace và quá trình thiết lập được lưu trữ. Trước khi tạo ra cửa sổ ảnh hay văn bản cho các workspaces, tất cả các cửa sổ ảnh, văn bản cần được đóng lại.

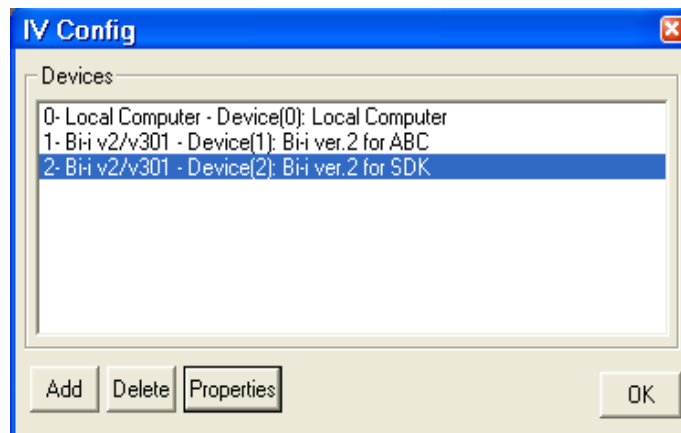
Khi mở workspaces, quá trình phê chuẩn cần xét duyệt nếu đường dẫn lưu trữ của workspaces khác với đường dẫn thực của tập workspaces. Quá trình rà soát bắt đầu khi lựa chọn nút Yes trong hộp thoại Confirmation. Khi rà soát, tất cả các tệp và thư mục trong workspaces sẽ được kiểm tra. Nếu tệp hoặc thư mục bắt đầu với cùng với đường dẫn gốc của workspaces, nó sẽ được thay thế bằng đường dẫn thực. Kỹ thuật rà soát: tệp workspaces và các tệp liên quan, thư mục có thể được di chuyển tới một vị trí khác an toàn hơn. Sự rà soát hoạt động chính xác nếu tệp workspaces được đặt trong thư mục mức cao nhất, tất cả các tệp phải được đặt trong thư mục đó hoặc là các thư mục con trong nó.

Chọn Save Workspaces từ File để lưu workspaces với tên hiện tại của nó hoặc Save Workspaces As để lưu workspaces với tên mới. Khi lựa chọn Close Workspaces từ File, workspaces hiện tại được đóng và các thiết lập được lưu trữ trong cấu hình.

IVConfig

IVConfig là thanh công cụ quản lý cấu hình các thiết bị trong hệ thống của bạn: hệ InstantVision và platform mà các thiết bị kết nối. Nó bao gồm tất cả các thông số cho quá trình thực hiện trên platform. Có một số loại khác nhau giữa các thiết bị và được hướng dẫn cụ thể trong hướng dẫn sử dụng của platform hiện tại. Ứng dụng dựa trên InstantVision có thể thực hiện trên thiết bị đơn giản nhất là Local Computer.

Thanh công cụ IVConfig đặt trên thanh điều khiển Control của máy tính. Chọn 'Control Panel Options' trong WindowsXP, hộp thoại chính của IVConfig cho thấy danh sách các thiết bị cần cấu hình. Mỗi thiết bị xác định bởi số hiệu. Số hiệu này sử dụng khi chạy ứng dụng trên platform chỉ định: chỉ chạy thiết bị có số hiệu được chọn, tất cả các thiết lập khác tạm bỏ qua. Nhiều thiết bị có cùng số hiệu có thể trở đến cùng một platform. Hộp thoại IV Config như trong hình 2.41.



Trong đó: Hình 2.41. Hộp thoại IV Config

<i>Add</i>	Thêm thiết bị mới, hộp thoại Add New Device xuất hiện
<i>Delete</i>	Xóa thiết bị đã lựa chọn.
<i>Properties</i>	Đặt đặc tính cho thiết bị lựa chọn, hiển thị hộp thoại Device Properties.
<i>OK</i>	Đóng IVConfig.

Ấn nút Add trên hộp thoại IV Config, hộp thoại Add New Device xuất hiện như trong hình 2.42.



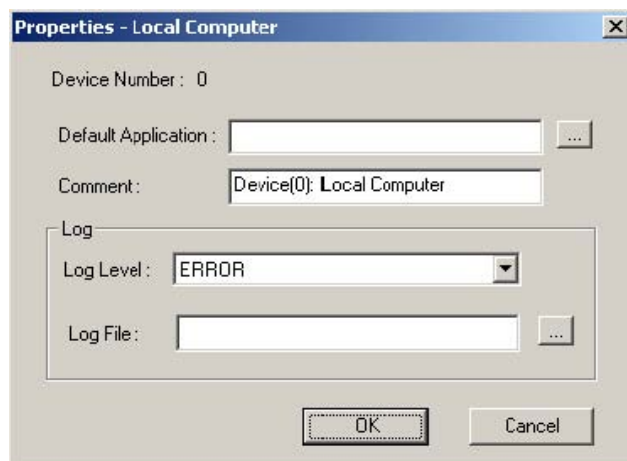
Hình 2.42. Hộp thoại Add New Device

Trong đó:

<i>Device Number</i>	Số hiệu của thiết bị khi chạy ứng dụng trên nó.
<i>Device Typ</i>	Local Computer và Bi-iV2/v301.

<i>OK</i>	Thêm thiết bị và trở về màn hình chính.
<i>Cancel</i>	Trở về màn hình chính mà không thêm thiết bị.

Để con trỏ ở vị trí của Local Computer trong danh sách trên IV Config, ấn nút Properties, hộp thoại Properties-Local Computer như trong hình 2.43.



Hình 2.43. Hộp thoại

Trong đó:

<i>Default Application</i>	Nó là tệp DLL của ứng dụng DLL. Nếu đường dẫn không chỉ định, tệp DLL sẽ tìm đến thư mục 'bin' trong thư mục cài đặt của InstantVision ISE.
<i>Comment</i>	Thêm chú giải xuất hiện trong ô hộp thoại.
<i>Log</i>	Cho phép mức driver logging cho mục đích dịch soát lỗi. Thông thường ô Log file để trống, không cho phép logging. Mức Log level chỉ liên quan nếu Log file đã lựa chọn.
<i>OK</i>	Lưu các đặc tính của thiết bị và trở về cửa sổ chính.
<i>Cancel</i>	Trở về cửa sổ chính mà không lưu các đặc tính của thiết bị.

Cài đặt và cho chạy phần mềm ứng dụng PILLS

Để thử nghiệm phần mềm PILLS ứng dụng công nghệ mạng nơ ron tế bào cho phân loại thuốc viên, cần thiết lập lại mô hình thử nghiệm chương trình trong phòng thí nghiệm như trong hình 2.22 và 2.23

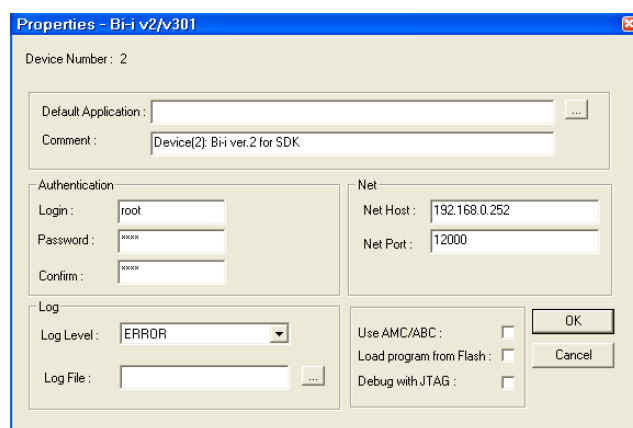
Sau khi thiết lập lại mô hình trong phòng thí nghiệm, ta cần tiến hành các bước tiếp theo:

- Cài đặt và kết nối Bi-i với PC qua mạng Ethernet cũng như cách kiểm tra Bi-i đã hoạt động tốt hay chưa?
- Chạy chương trình PILLS đã xây dựng trên ứng dụng chủ IVRun trên PC
- Cho hệ thống hoạt động và quan sát các kết quả đạt được tiến hành trong phòng thí nghiệm.

• Đặt cấu hình thiết bị Bi-i

Nhiều Bi-i có thể cùng nối vào một mạng và thiết lập cấu hình cho Bi-i bằng nhiều cách khác nhau. Trước khi sử dụng Bi-i, cần vào mục cấu hình thiết bị Bi-i để trở đến Bi-i cần sử dụng. Sau khi cài đặt phần mềm Bi-i, một số thiết bị mặc định được đặt trong bảng cấu hình thiết bị nhưng vị trí mạng phải đặt bằng tay.

Đặt cấu hình cho Bi-i có thể sử dụng IVConfig, đó là thành phần của InstantVision ISE. Khởi động IVConfig từ thanh điều khiển máy tính của bạn. Một danh sách các thiết bị xuất hiện, thiết bị có thể thêm vào, xóa đi và đặc tính của thiết bị cũng được thiết lập lại. Lựa chọn một thiết bị Bi-i trong danh sách của IVConfig và nhấp nút Properties để đặt các thông số cho chúng. Đặc tính thiết bị Bi-iV2 trong IVConfig được đặt như trong hình 2.44.



Hình 2.44: Đặc tính thiết bị Bi-iV2 trong IVConfig

Ở đây, địa chỉ IP của Bi-iV2 trong mạng là 192.168.0.252. Khi chạy ứng dụng phát triển với Bi-i SDK thì không lựa chọn ‘Use AMC/ABC’, ‘Debug with JTAG’ và ‘Load program from Flash’.

- **Kiểm tra Bi-i**

Cách đơn giản nhất để kiểm tra Bi-i là chạy một chương trình đơn giản AMC. Thiết bị cần được đặt cấu hình cho ABC (compiled AMC). Hiện thị đặc tính của thiết bị trong IVConfig và kiểm tra lại:

- Đặt “Authentication” hoàn tất (có thể truy nhập như 'root' với mật khẩu 'pass' trừ phi giá trị mặc định đã bị thay đổi).
- Vùng Network host trở tới Bi-I và network port đặt là 12000.
- Chọn 'ABC file required'.
- Không chọn 'Debug with JTAG', không chọn 'Load program from Flash'.

Sau đó chạy AMC edit từ thực đơn khởi động và hoàn thành các bước sau:

B1. Chọn mục thiết bị từ thực đơn Project lựa chọn thiết bị đã đặt cấu hình trong trong hộp thoại hiện ra, ấn nút Select khi hoàn thành.

B2. Mở ứng dụng AMC với mục Open của thực đơn File. Ví dụ, 'Samples\AMC\Base>HelloWorld>HelloWorld.amc' trong thư mục cài đặt của phần mềm Bi-i sẽ gửi một thông điệp.

B3. Chọn mục Execute của thực đơn Project. Nó sẽ dịch và gọi IVRun để thực hiện. Nếu HelloWorld.amc đang chạy, dòng chữ 'Hello World!' sẽ hiện ra trong IVRun.

- **Chạy chương trình PILLS**

Để chạy phần mềm ứng dụng công nghệ mạng nơ ron tế bào cho phân loại thuốc viên PILLS chuyển động tốc độ cao cần làm theo các bước sau:

1. Mở tệp dự án PillDemo.pjt với Code Composer Studio v3.1 đã xây dựng cho hệ thống Bi-iV2.
2. Build dự án đã mở.

Nếu đã có tệp *PillDemo.out* (tệp được tạo ra sau khi tiến hành compiler trên phần mềm Code Composer Studio v3.1, có thể sao chép tệp này từ đĩa CD cài đặt vào máy tính) thì bắt đầu từ bước 3.

3. Sử dụng thiết bị Bi-iV2 với IVConfig và đặt các vùng trên Device Properties cho phù hợp.
4. Mở phần mềm ứng dụng chủ IVRun trên PC.
5. Trên IVRun, mở tệp ứng dụng PillDemo.out sử dụng chức năng File/Open Application (**Ctrl+A**).
6. Cho phép cập nhật thư mục.
7. Chọn thiết bị trong thư mục Settings/Run: Bi-iV2 for SDK.
8. Sử dụng chức năng File/Save Workspace lưu workspace để tránh lặp lại các bước trước đó khi mở workspace lần sau.
9. Ấn nút Run trên thanh công cụ hoặc chọn chức năng IV/Run cho phép chạy ứng dụng.

2.3. Thí nghiệm kiểm tra nhanh đại ốc đường sắt sử dụng công nghệ CNN

2.3.1. XÂY DỰNG MÔ HÌNH VÀ THUẬT TOÁN

Ý tưởng của thử nghiệm này nhằm thử nghiệm khả năng phát hiện một vài lỗi của đường ray trong quá trình tàu chạy (on-line). Đây là bài toán rất khó vì đòi hỏi nhiều công nghệ xử lý tín hiệu và hình ảnh tốc độ cao trong môi trường rung xóc.

Nếu ta sử dụng camera lắp trên tàu để kiểm tra ốc tà vẹt thì đòi hỏi tốc độ xử lý ảnh của camera phải rất cao. Giả sử tàu chạy với tốc độ 80km/giờ tức là 22.2 m/s. Mỗi frame ảnh thu 20cm. Một mét cần thu 5 frame ảnh. Như vậy trong một giây camera cần thu 111 frame ảnh.

Để thử khả năng xử lý ảnh nhanh công nghệ CNN và hệ thống Bi-I trong hệ thống kiểm tra tự động chất lượng ốc giữa các thanh ray, chúng tôi xây dựng mô hình để thử nghiệm gồm một đĩa quay với tốc độ 3000vòng/phút hay 50vòng/giây. Ta gắn 4 ảnh ốc đường ray trên đĩa quay. Khi ống kính camera Bi-I chiếu vào một ảnh thì thời gian tồn tại của ảnh này trước ống kính còn 1/4 so với gắn 1 ảnh hay nói cách khác tốc độ tương đương 200 fps đủ thỏa mãn yêu cầu thực tế > 111 fps . Trên 4 ảnh có một ảnh không có mũ ốc. Thí nghiệm phải nhận biết được ảnh này khi đĩa quay 3000 vòng phút (tương đương tốc độ đoàn tàu 145km/giờ).

Thuật toán CNN và các mẫu trọng số sử dụng

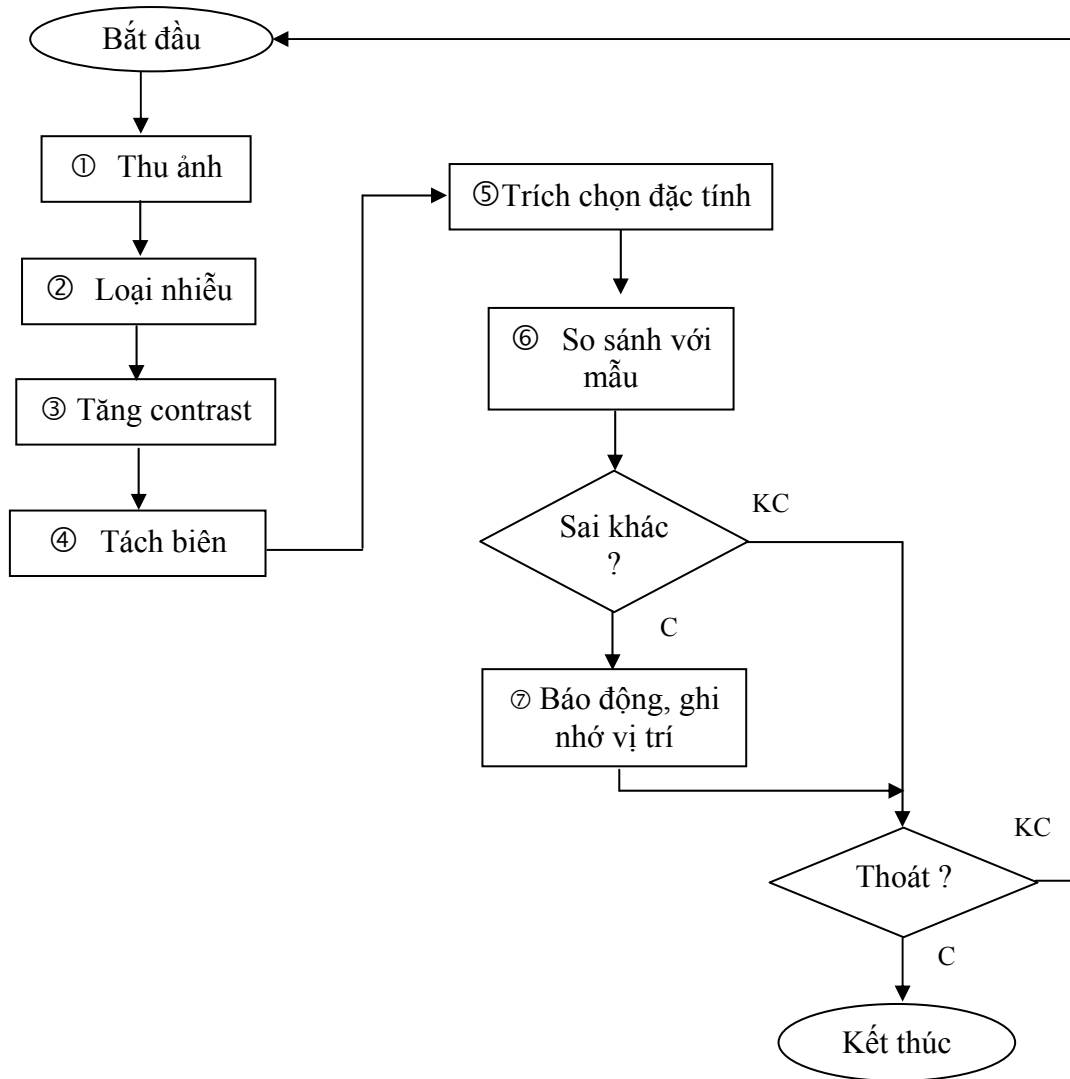
Các ảnh thu được từ camera của Bi-I cũng là các ảnh số. Các bước xử lý ảnh số truyền thống thường có hai phần: phần tiền xử lý và phần xử lý. Các công việc trong phần tiền xử lý ảnh gồm:

- + Làm tron ảnh-Khử nhiễu-Tăng độ tương phản
- + Phát hiện đặc tính biên, phân vùng-Trích chọn đặc tính và
- + Nhận dạng, phân lớp, cho ra quyết định...

Tùy theo ảnh đầu vào và mục đích xử lý mà số bước công việc này có thể có hoặc không (Một ví dụ đơn giản: khi ảnh thu được đã có chất lượng tốt thì ta cũng không cần thực hiện bước tiền xử lý tăng chất lượng của ảnh nữa)

Để thực hiện xử lý ảnh dùng CNN chúng tôi cũng tiến hành theo hai bước này. Điểm khác biệt ở đây là thay cho việc tính toán từng bit trên mỗi bước thì chúng tôi thực hiện đưa các bộ mẫu (template) cần thiết vào các bộ xử lý CNN để thực hiện tính toán song song bằng một lệnh duy nhất. Như vậy rõ ràng tốc độ xử lý có thể tăng lên hàng chục nghìn lần.

- Thuật toán của chương trình thử nghiệm trên Bi-IV2



Hình 2.45. Thuật toán xử lý

- Chi tiết các bước và mẫu sử dụng:

Các bước từ 1 đến 5 được thực hiện trên bộ xử lý CNN (xem hình 2.46). Một số mẫu được sử dụng

a)
$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \hat{D} = \begin{bmatrix} d & d & d \\ d & 0 & d \\ d & d & d \end{bmatrix} \quad z = -1$$

$d \quad 0.2 \quad 0.1 \quad 2.5 -1$

b)
$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.3 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -0.3 & -0.3 & -0.3 \\ -0.3 & 2.4 & -0.3 \\ -0.3 & -0.3 & -0.3 \end{bmatrix} \quad z = \boxed{0}$$

c)
$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad z = \boxed{-1}$$

Hình 2. 46. Một số mẫu được sử dụng a) mẫu lọc b) mẫu tăng contrast c) mẫu dò biên

Bước trích chọn đặc tính là một trong các bước quyết định của chương trình. Không có khuôn mẫu chung nào cho bước này vì đối tượng quan tâm trong ảnh khác nhau tùy theo yêu cầu của chương trình. Chương trình thí nghiệm đặt mục tiêu kiểm tra trong frame ảnh có mũ ốc hay không có mũ ốc?

Thuật toán thực hiện

Để giải quyết bài toán này trong thực tế rất khó khăn vì như chúng ta đã biết các đai ốc giữ các thanh tà vẹt có thể bị các lá cây, rác che khuất...dẫn tới các ảnh của chúng không còn giữ được hình dạng đặc trưng nguyên thủy. Tất nhiên khi ứng dụng vào thực tế cũng có thể kết hợp các biện pháp khác để tạo điều kiện quan sát ảnh tốt hơn. Ở đây, thực hiện ở mức phòng thí nghiệm; để kiểm tra khả năng dùng Bi-IV2 để phát hiện được đối tượng có mất mũ ốc không ngay trong khi tàu chạy nên chúng tôi giả định là các ảnh còn giữ nguyên được hình dạng nguyên thủy. Vì vậy có thể dùng phương pháp tính toán đặc điểm, số liệu thống kê của đối tượng (feature extraction) để đi tới kết luận. Kích thước ảnh khi thu bằng camera Bi-IV2 là 128x128, ta gọi ảnh này là ảnh P. Vùng ảnh chứa đối tượng cần theo dõi (mũ ốc) là một vùng nhỏ nằm trong P. Thực hiện tiếp thao tác tách vùng cần theo dõi với kích thước 40x40 để thu được ảnh mới P1 mà ảnh này chỉ có duy nhất đối tượng nếu nó tồn tại. Nếu đối tượng tồn tại bình thường (không mất mũ ốc) thì sau những khoảng cách nhất định (khoảng cách giữa các thanh tà vẹt -1m) phải thấy nó trên P1. Nếu sau 2 lần khoảng cách này mới thấy đối tượng trên P1 thì có thể kết luận là bị mất mũ ốc.

Việc trích chọn đặc tính sử dụng trên cơ sở tính toán các trục x,y (Major axis and Minor axis) là các trục của hình elip ảo trên đối tượng. Ngoài ra còn có thể sử dụng phương pháp thống kê điểm ảnh để thực hiện các bước ⑤ và ⑥ của thuật toán đã nêu. Nội dung của phương pháp này là tính số điểm đen (trắng) của đối tượng trong P1. Nếu sau 2 lần khoảng cách giữa các thanh tà vẹt mà không thu được ảnh đầy đủ của đối tượng trong P1 mà thể hiện là tổng số điểm đen (trắng) thực tế nhỏ hơn tổng số điểm đen (trắng) trong khoảng bình thường thì có thể kết luận là mất ốc.

Kết quả thực nghiệm

Thực nghiệm đã được thực hiện với các thông số: Đĩa quay 3000vg/ph. Tốc độ thu ảnh 500 fps, ảnh đen trắng mô hình, ánh sáng chiếu lên ảnh bình thường bằng đèn 100W ở khoảng cách 30cm. Camera Bi-IV2 dùng ống kính 25 1:1.4. Chương trình thực hiện bằng ngôn ngữ lập trình Code Composer Studio 3.1 do hãng Texas cung cấp. Chương trình thực hiện đã phát hiện được các ảnh mất mũ ốc như yêu cầu đề ra.. Tuy nhiên đây chỉ là thí nghiệm khả năng xử lý ảnh tốc độ cao của Bi-I đạt được tốc độ chuyển động của tàu. Để ứng dụng được vào bài toán thực tế còn phải giải quyết nhiều vấn đề liên quan cả về công nghệ lẫn phương pháp xử lý tổng hợp.

Đây chỉ là một trong vô số lĩnh vực mà tiềm năng sử dụng công nghệ CNN cho phép tháo gỡ các khó khăn không thể giải quyết gặp phải khi sử dụng các máy tính xử lý lệnh tuần tự. Trong giai đoạn tới đây việc nghiên cứu và ứng dụng công nghệ này tại Việt Nam là rất cần thiết và có ý nghĩa rất lớn, đặc biệt trong lĩnh vực công nghiệp, quốc phòng và an ninh.

2.3.2 HƯỚNG DẪN SỬ DỤNG THÍ NGHIỆM KIỂM TRA NHANH ĐAI ỐC

Giới thiệu

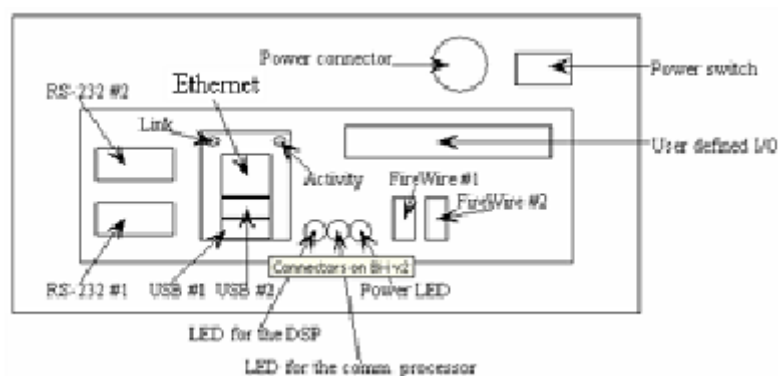
Phần mềm thử nghiệm kiểm tra nhanh đai ốc đã được phát triển và nạp trong đĩa **CD** “*các mô hình và chương trình thử nghiệm công nghệ xử lý ảnh tốc độ cao phục vụ cho nghiên cứu và đào tạo*” của nhiệm vụ nghị định thư.

Các bước chuẩn bị

- Kết nối Bi-i V2 và máy tính PC

Thiết bị Bi-i V2 được nối với một máy tính PC qua Ethernet thông qua một hub. Khi nối vào mạng thiết bị Bi-i V2 sẽ có một địa chỉ IP xác định. Việc thiết lập địa chỉ cho Bi-i V2 được thực hiện qua IPconfig. Chẳng hạn thiết bị sử dụng địa chỉ IP 192.168.0.251.

Quan sát các đèn phía sau Bi-i V2 để kiểm tra dây kết nối mạng



Hình 2.47 Mặt sau của thiết bị Bi-I

Phần mềm

Chương trình thí nghiệm xác định mũ ốc bị mất chạy trên Bi-i V2, để hiển thị các thông tin sử dụng máy tính PC cài hệ điều hành Windows 2000 hoặc XP. Trên máy tính cần cài gói phần mềm Bi-i:

- + Gói phần mềm sau khi cài đặt hỗ trợ cho lập trình AMC, lập trình SDK và lập trình API.
- + Phần mềm Bi-iconfig cho phép thiết lập các thông số của Bi-i trong hệ thống
- + Phần mềm IVRun để chạy chương trình giao diện trên máy tính PC

Mô hình thí nghiệm

Mô hình thử nghiệm được mô tả trong hình 2.48

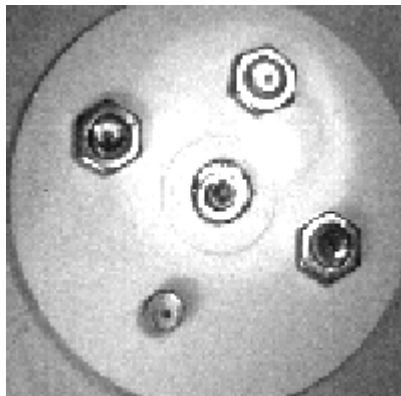


Hình 2.48: Mô hình thí nghiệm kiểm tra đại ốc

- + Để thực hiện thí nghiệm cần có một thiết bị đĩa quay để gắn đĩa mang 4 bu lông. Có 3 bu lông có mũ ốc và 1 bu lông không có mũ ốc.
- + Đèn dây tóc chiếu sáng 100W
- + 1 Thiết bị Bi-i
- + 1 Hub

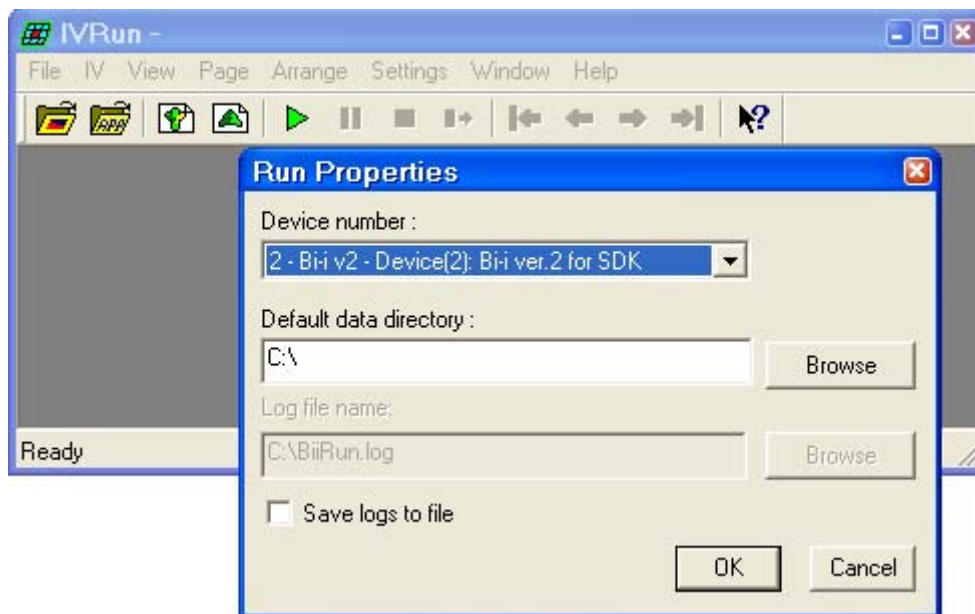
Thực hiện chương trình

- + Thiết bị Bi-i đặt cách thiết bị quay khoảng 35 cm
- + Bật các đèn chiếu sáng
- + Khởi động IVRun
- + Chạy file chương trình Program8-12.cwp
- + Hiệu chỉnh thiết bị Bi-i để thu được ảnh như hình 2.46
- + Chọn thiết bị Bi-i qua mục “Run Properties\Run”
- + Thiết lập hình dáng các cửa sổ của chương trình thực hiện qua thuộc tính “Setting\display”. Thuộc tính này cho phép điều chỉnh kích cỡ, màu sắc các cửa sổ hiển thị, kích cỡ màu sắc chữ (text) trên cửa sổ



Hình 2.49: Chính vị trí Bi-i để thu được ảnh đầy đủ

- + Bật công tắc thiết bị quay để đĩa mang các mũ ốc chuyển động
- + Điều chỉnh tốc độ đĩa quay và quan sát chương trình hoạt động



Hình 2.50: Hộp thoại Run

Kết quả xử lý hiện trên màn hình như hình 2.51 sau:



Hình 2.51: Kết quả xử lý kiểm tra đai ốc trên màn hình

2.4. Thử nghiệm khả năng thu ảnh nhanh các sự kiện thay đổi đột ngột bằng thí nghiệm nổ bong bóng

2.4.1. Mô tả thí nghiệm

- Các thành phần trong hệ thí nghiệm

Mục đích của hệ thí nghiệm nhằm phân tích quá trình nổ bóng trong phòng thí nghiệm, sử dụng khả năng xử lý ảnh vượt trội của máy tính đa năng dựa trên công nghệ mạng neuron tế bào CNN-UM. Thông thường khi bóng nổ, bằng mắt thường hoặc bằng các thiết bị thu nhận ảnh truyền thống ta sẽ không thể theo dõi được diễn biến quá trình nổ của quả bóng, do quá trình này xảy ra rất nhanh, chỉ vài mili giây. Trong khi đó, sử dụng công nghệ CNN ta không chỉ ‘quay chậm’ lại từng chi tiết của quá trình xảy ra mà còn kịp thời xử lý những ảnh đó với tốc độ vài chục nghìn ảnh/giây. Hệ thí nghiệm này bao gồm máy tính đa năng Bi-I v2 làm nhiệm vụ bắt giữ và xử lý ảnh thu được từ môi trường bên ngoài. Máy tính Pentium IV 2.1 GHz làm nhiệm vụ lưu giữ ảnh, xử lý tập lệnh và truyền thông với máy tính đa năng CNN. Một bàn thí nghiệm trên đó có đặt các quả bóng sẽ được kích nổ.

Phần quan trọng nhất của thí nghiệm là máy tính đa năng CNN Bi-I v2, ta sẽ tìm hiểu về vai trò của siêu máy tính này trong thí nghiệm.

Vai trò của máy tính đa năng Bi-I V2 trong thí nghiệm

Ưu điểm của chip Neuron tế bào đã thúc đẩy sự ra đời của các thế hệ máy tính xử lý ảnh đáp ứng tiêu chuẩn công nghiệp như máy tính Alladin sử dụng chip ACE4k kích thước 64 x 64, máy tính Bi-I của hãng Analogic Computer Ltd sử dụng chip ACE16k kích thước 128 x 128... Các máy tính này được tích hợp một bộ xử lý tín hiệu số chất lượng cao DSP. Bộ DSP được sử dụng vì 2 lý do, trước hết DSP cung cấp dữ liệu cho chip CNN và điều khiển hoạt động của chip này. Mặt khác, DSP đóng vai trò quan trọng khi nhiệm vụ xử lý ảnh chứa một số toán hạng logic. Sau khi toàn bộ quá trình tính toán tiền xử lý phức tạp (bao gồm một số lượng lớn toán hạng xử lý ảnh) được thực hiện bởi chip CNN, DSP sẽ hoàn thành nốt nhiệm vụ còn lại. Tức là, chip CNN sẽ lọc ra khoảng 1% ảnh cần quan tâm, và DSP sẽ chỉ làm việc trên phần dữ liệu được rút gọn đáng kể này. Hai bộ xử lý chất lượng cao được tích hợp để tạo ra một hệ thống thị giác cực mạnh, có khả năng tính toán ảnh thời gian thực trong các ứng dụng có yêu cầu cao. Bài báo này sẽ giới thiệu về máy tính đa năng CNN Bi-I v2 và khả năng xử lý ảnh tốc độ cao của nó.

Chương trình phần mềm viết trên ngôn ngữ Bi-I SDK

Bi-I v2 hỗ trợ ba môi trường lập trình khác nhau tùy theo mức độ phức tạp của ứng dụng.

- Ngôn ngữ AMC (Analogic Macro Code) là ngôn ngữ lập trình thông thường của Bi-I đối với các ứng dụng đơn giản. Ngôn ngữ thường sử dụng khi thực hiện các thuật xử lý ảnh tương đối đơn giản, ví dụ kiểm tra các tính năng của bộ xử lý ACE16K. Tất cả các công cụ

phần mềm cần thiết cho lập trình AMC được cài đặt với phiên bản Standard hoặc Development của phần mềm Bi-I. Tất cả các chức năng của Bi-I có thể tìm hiểu qua chương trình AMC.

- Bi-I API là chương trình lập trình giao diện được sử dụng trên máy tính chủ để viết đoạn mã làm việc với Bi-I. Chương trình trên máy chủ có thể gọi các chương trình trên Bi-I và trao đổi dữ liệu giữa máy tính chủ và Bi-I. Chương trình trên máy chủ mặc định là chương trình *Bi-IRrun*, đi kèm với phần mềm Bi-I. Sử dụng chương trình Bi-I API nếu muốn thay đổi giao diện người dùng hoặc nếu chương trình cần phải thực hiện các nhiệm vụ khác mà ngôn ngữ AMC không thực hiện được (như làm việc với cơ sở dữ liệu hoặc các thiết bị ngoại vi khác). Bi-I được sử dụng với bất kỳ môi trường lập trình nào hỗ trợ thư viện động.

- Bi-I SDK là một thư viện phần mềm lập trình C++ được sử dụng cho phát triển các ứng dụng Bi-I. Sử dụng SDK nếu ứng dụng lớn, phức tạp. Bi-I SDK được thiết kế để sử dụng cùng với CodeComposer Studio của hãng Texas Instruments. Bi-I SDK làm việc cùng thư viện xử lý ảnh InstantVision của Analogic.

Thư viện phần mềm InstanceVision bao gồm các modul sau:

- Cấu trúc cơ sở dữ liệu (BaseData)
- Thư viện xử lý ảnh (ImgLib)
- Thư viện phân loại theo đặc trưng (FCLib)
- Thư viện bám đa mục tiêu (MTTLib)

Các kiểu đối số trong ngôn ngữ lập trình:

Có hai nhóm kiểu dữ liệu chính của ngôn ngữ SDK:

- Ảnh và mẫu trên ACE16k
- Đối số trên DSP

Ảnh và mẫu trên ACE16k

C_TEM	Bộ nhớ chứa các mẫu
C_LLM	Bộ nhớ ảnh nhị phân
C_LAM	Bộ nhớ ảnh mức xám

Các đối số trên DSP

- Kiểu vô hướng. Độ lớn của các kiểu vô hướng là 4 byte.

M_GLM	Logical type scalar (0 - 1).
M_GAMB	Byte type scalar. Unsigned integer (0 - 255).
M_GAM	4-byte signed integer scalar.
M_GAMF	4-byte floating-point scalar.

- Kiểu vector

M_GLM_1D	Logical type 1D vector. The range of its elements is 0 - 1.
----------	---

M_GAMB_1D	Byte type 1D vector. The range of its elements is 0 - 255 (unsigned integer).
M_GAM_1D	4-byte signed integer 1D vector.
M_GAMF_1D	4-byte floating-point 1D vector.

- Kiểu mảng 2D (Ma trận).

M_GLM_2D (M_LLM) A two dimensional array of logical values (0 or 1). This is identical to M_LLM. M_LLM means the binary image.

M_GAMB_2D (M_LAM) A two dimensional array of unsigned integers (0-255). This is identical to M_LAM. M_LAM means the gray scale image.

M_GAM_2D A two dimensional array of 4-byte signed integers.

M_GAMF_2D A two dimensional array of 4-byte floating-point numbers.

M_RGB Color image. Actually, these are three M_LAM array with one header.

- M_TEM: Bộ nhớ mẫu (floating-point scalar array with 32 elements).
- Các kiểu khác.

M_ADR Memory address without type.

M_STR Address of a string.

STR String

LABEL Name of an existing label in the AMC code.

NIL Empty data, indicates that the operand is not given.

ACE16k có 32 bộ nhớ mẫu TEM1-TEM32, 8 bộ nhớ ảnh xám LAM1-LAM8 và hai bộ nhớ ảnh nhị phân LLM1-LLM2.

Trong bộ nhớ DSP, kích thước ảnh được mặc định là 128 x 128.

Ảnh và các mảng 2D

Tất cả các ảnh và mảng 2D bắt đầu với một header kích thước 5 word (20 byte) mô tả kiểu và kích thước của dữ liệu. Cấu trúc của header như sau:

1. Kiểu mảng
2. Số lượng các phần tử trong một hàng
3. Số lượng hàng
4. Kích thước của hàng theo word
5. Kích thước của mảng nếu không có header

Trong Bi-I V2 có ba kiểu ảnh được sử dụng:

- Kiểu ảnh nhị phân: (M_LLM hoặc M_GLM_2D). Mỗi pixel chiếm 1 bit Giá trị của kiểu mảng trong header là 0.

- Kiểu ảnh mức xám (M_LAM hoặc M_GAMB_2D) với 8 bit mức xám, mỗi pixel chiếm một byte. Giá trị của kiểu mảng trong header là 1.

- Kiểu ảnh màu (M_RGB). Cấu trúc dữ liệu này chứa 1 header ảnh và 3 ảnh xám cho 3 kênh màu Red, Green và Blue. Giá trị của kiểu mảng trong header là 2. Giá trị của kích thước ảnh trong header nghĩa là kích thước ảnh của một kênh màu.

Ví dụ tính kích cỡ của một ảnh xám 128 x 128:

$$\begin{aligned} \text{Image size in bytes} &= 5 * 4 \text{ (header)} + \\ &128 \text{ (height)} * ((128 \text{ (width)} + 3) / 4) \text{ (row size in words)} * 4 = 20 + \\ &128 * ((128 + 3) / 4) * 4 = 16404 \text{ bytes} \end{aligned}$$

Vector

Vector thực chất là mảng một chiều. Cấu trúc của header khác so với header của mảng hai chiều, do kích thước của nó là 3 word (12 byte)

1. Kiểu vector
2. Số lượng phần tử trong mỗi vector
3. Kích thước của vector không có header

Ví dụ kích thước của vector chứa 20 phần tử boolean:

$$((20 \text{ (number of elements)} + 31) / 32) \text{ (size in words)} * 4 = 16 \text{ bytes}$$

Các mẫu:

Mẫu là một mảng đơn giản, chứa 32 giá trị kiểu float (128byte).

Làm việc với ảnh

Có một số phương pháp để đưa một ảnh vào bộ nhớ DSP:

- Đọc ảnh từ máy tính
- Thu ảnh từ một ứng dụng
- Thu ảnh với cảm biến Ibis CMOS
- Thu ảnh với chip ACE16k
- Đọc các chuỗi ảnh từ file AVI
- Thu ảnh từ thiết bị thu thập Frame grabber trong máy tính
- Tạo một ảnh trong DSP

Ví dụ đọc ảnh từ máy tính và lưu vào bộ nhớ DSP :

```
;Loads color image from the host computer to the DSP memory.  
host.load.pic rgb.bmp $0x80010000 rgb  
; địa chỉ để lưu vào bộ nhớ của DSP là $0x80010000  
;Displays the image to the host application.  
host.display $0x80010000 1
```

Bộ nhớ trên chip ACE16k

ACE16k nối trực tiếp tới DSP nên dữ liệu về ảnh được lấy trực tiếp từ DSP. Toàn bộ ảnh được lấy từ DSP được thực hiện qua các biến được định nghĩa trước:

Bộ nhớ ảnh xám: LAM1 – LAM8

Bộ nhớ ảnh số : LLM1 – LLM2

Bộ nhớ mẫu : TEM1 – TEM32

Chương trình Bi-I SDK

Bi-I SDK là một tập hợp thư viện C++ được sử dụng để lập trình Bi-I. Một số phần của Bi-I SDK dựa trên các lớp được định nghĩa trong modul BaseData của thư viện InstantVision. Bi-I SDK được thiết kế để sử dụng cùng với CodeComposer Studio của hãng Texas Instruments.

Bộ nhớ:

Bi-I v2 được xây dựng với một DSP TMS320C6415 có 1MB bộ nhớ bên trong, có thể sử dụng như bộ nhớ dữ liệu hoặc bộ nhớ chương trình. Dung lượng bộ nhớ ngoài là 16MB. Bộ nhớ bên ngoài có thể sử dụng tự do để lưu trữ dữ liệu hoặc chương trình, tuy nhiên chương trình có thể chạy chậm hơn nhiều. Nói chung, đoạn mã thực hiện mất nhiều thời gian và dữ liệu của nó được lưu giữ ở bộ nhớ trong. Việc sao chép dữ liệu đến bộ nhớ bên trong, thực hiện tính toán và gửi kết quả ngược lại đến bộ nhớ bên ngoài có thể nhanh hơn tính toán trực tiếp trên dữ liệu ở bộ nhớ ngoài. Các vùng bộ nhớ của hệ Bi-I v2 dưới dạng hexa như sau:

Bộ nhớ trong: 0x00000000 – 0x000FFFFFFF
Bộ nhớ ngoài: 0x80000000 – 0x80FFFFFFF

Truyền thông với máy tính chủ:

Một ứng dụng Bi-I có thể truyền thông với ứng dụng chạy trên máy chủ thông qua các yêu cầu (requests). Tất cả các yêu cầu được lưu giữ trong hàng đợi ở bộ nhớ của Bi-I. Một tiến trình yêu cầu bao gồm các quá trình sau:

- Ứng dụng Bi-I đặt yêu cầu lên hàng đợi
- Ứng dụng Bi-I gửi tín hiệu đến máy chủ thông báo rằng có một yêu cầu mới .
- Máy tính chủ sẽ đọc yêu cầu từ hàng đợi
- Yêu cầu được xử lý trên máy chủ
- Ứng dụng trên máy chủ gửi một tín hiệu đến ứng dụng Bi-I để chỉ ra rằng yêu cầu đã được thực hiện.

Các chế độ truyền thông:

Bi-I SDK hỗ trợ hai kiểu truyền thông giữa ứng dụng Bi-I và ứng dụng trên máy chủ: đó là chế độ truyền thông đồng bộ và chế độ truyền thông dị bộ.

Chế độ truyền thông đồng bộ là kiểu truyền thông mặc định. Ứng dụng Bii bị treo tạm thời từ khi Bii gửi một yêu cầu đồng bộ, cho đến khi nhận được thông báo của máy tính chủ là đã xử lý xong yêu cầu. Trong kiểu truyền thông này, hàng đợi chỉ chứa duy nhất một yêu cầu tại một thời điểm.

Chế độ truyền thông kiểu dị bộ là kiểu truyền thông được phát triển, cho phép mã lệnh vẫn được kích hoạt trên Bii trong khi quá trình xử lý yêu cầu đang được thực hiện trên máy chủ. Ứng dụng Bi-I không phải đợi tín hiệu trả lời từ máy chủ mà các yêu cầu được xếp hàng trong hàng yêu cầu.

Tất cả quá trình xử lý truyền thông được điều khiển bởi lớp THostStream. Có thể đặt theo một số chế độ đó là chế độ AVI, Ảnh, các file Ảnh, các file Mẫu và Dữ liệu, tùy theo kiểu file. Ở chế độ chuẩn, THostStream gửi và nhận dữ liệu đến và từ ứng dụng trên máy chủ qua chương trình giao diện BiiAPI. THostStream cũng có thể trao đổi dữ liệu với chương trình giao diện Matlab trên máy chủ hoặc với card xử lý ảnh theo chế độ của card xử lý ảnh.

Tất cả các phiên bản của Bi-I đều chứa một chip ACE16k. ACE16k chứa một cảm biến ảnh mức xám CMOS độ phân giải thấp (128 *128) và một mảng xử lý tín hiệu tương tự. (Một bộ xử lý gắn với một điểm ảnh mà không cần chuyển đổi A/D). Tất cả các đơn vị xử lý được liên kết đến các bộ xử lý liền kề với trọng số nhất định. Mảng xử lý này rất có hiệu quả trong một số phép xử lý ảnh hơn các bộ xử lý truyền thống, bởi vì các công việc này xử lý toàn bộ ảnh trong cùng một thời điểm.

Giao diện vào/ra:

ACE16k có giao tiếp số 32bit, cho phép chuyển đổi các ảnh và các mẫu giữa ACE16k và DSP.

Bên trong bộ nhớ ACE16k chứa:

- 8 bộ nhớ ảnh xám (C_LAM1 – C_LAM8)
- 2 bộ nhớ ảnh số (C_LLM1 – C_LLM2)
- 32 bộ nhớ mẫu (C_TEM1 – C_TEM32)

Các ảnh được chuyển giữa hai bộ nhớ bên trong của ACE16k. Cả nguồn và đích đều có thể là ảnh nhị phân hoặc ảnh xám.

InstanceVision là tên gọi chung của các thư viện phần mềm sau:

- Cấu trúc cơ sở dữ liệu (BaseData)
- Thư viện xử lý ảnh (ImgLib)
- Thư viện phân loại theo đặc trưng (FCLib)
- Thư viện bám đa mục tiêu (MTTLib)

Các thư viện InstantVision có thể sử dụng với bất kỳ phần cứng nào chứa DSP TMS320C62xx hoặc TMS320C64xx từ Texas Instruments.

Cấu trúc cơ sở dữ liệu mô tả các kiểu cấu trúc dữ liệu, cung cấp môi trường cơ sở của InstantVision. Thư viện này bao gồm các kiểu và cấu trúc dữ liệu khác nhau như bit/byte/RGB/int/float ..., các ma trận các phương pháp chuyển đổi giữa các cấu trúc và ứng dụng. Các thành phần khác của thư viện đóng gói dựa trên thư viện này.

Thư viện xử lý ảnh là một tập hợp các lớp và các đường dẫn xử lý ảnh tổng quát. Thư viện này bao gồm các toán hạng xử lý ảnh nhị phân và ảnh xám, các chương trình tính đặc trưng, thống kê và một lớp để xử lý ảnh động.

Thư viện phân loại đặc trưng là tập hợp các phương pháp phân loại và tìm kiếm đặc trưng khác nhau, các chương trình nhận dạng, dựa trên quá trình học offline...

Thư viện bám nhiều mục tiêu có thể sử dụng cho thuật toán thích nghi để bám vài mục tiêu thậm chí trong môi trường nhiễu.

Một số lệnh Instant Vision sử dụng trong chương trình thử nghiệm thu ảnh bong bóng

1. Bắt giữ ảnh đầu vào

Cảm biến quan học sử dụng ánh sáng tạo ra dòng điện giữa lớp tiếp giáp N/P để giải phóng năng lượng cho một tụ điện mà đã được nạp trước đó. Giá trị nạp này thường là 195, khi thay đổi giá trị nạp này, độ tương phản của ảnh sẽ thay đổi.

Ảnh đầu ra từ các cảm biến được lưu giữ hoặc là ở trong bộ nhớ ảnh xám, hoặc ảnh nhị phân. Khi lưu giữ trong bộ nhớ ảnh nhị phân, ảnh sẽ được lấy ngưỡng 128 mức xám. Có hai kiểu dữ liệu thành phần của hàm bắt giữ ảnh trong lớp TACE.

- . Thời gian bắt giữ (mặc định là 500 microgiây)
- . Giá trị nạp trước (mặc định là 195)

Ví dụ :

```
TACE ace;
//Setting of capturing parameters
ace.Set_Int_Time(1000); //Integration time = 1 millisecond
ace.Set_Precharge(195); //Precharge value = 195
//Capturing to C_LAM1
ace.CaptureOpt(C_LAM1);
//Capturing to C_LLM1
ace.CaptureOpt(C_LLM1);
```

2. Lệnh đọc các ảnh từ một chuỗi video (AVI)

Toán hạng >> của lớp TvideoIn đọc các frame ảnh từ file AVI.

Ví dụ :

```
TByteMatrix ByteMatrix;
TVideoIn VideoIn("sample.avi");
for(int i=0; i<VideoIn.GetLength();i++)
{
VideoIn >> ByteMatrix;
}
```

3. Kiểu dữ liệu TByteMatrix

Lớp TbyteMatrix được khai báo trong TbyteMatrix.h. Ma trận này thể hiện ảnh mức xám. Các phần tử của ma trận này có giá trị từ 0 đến 255. Mỗi phần tử lưu giữ

Ví dụ:

```
// The ByteMatrix is a TByteMatrix object.
unsigned int length = ByteMatrix.DataSizeByte();
TBElem * p_element = (TBElem*)ByteMatrix.Data();
//Processes all elements of the matrix.
for(unsigned int i=0; i<length; i++)
{
```

```

if(*p_element < 205) *p_element = *p_element + 50;
else *p_element = 255;
p_element++;
}

```

4. Lệnh ghi một ảnh vào một chuỗi video (AVI)

Toán hạng << của lớp TVideoOut tạo ra một ảnh từ lớp lưu giữ ảnh và gán nó vào cuối của một file video.

Ví dụ

```

TByteMatrix ByteMatrix(320,240);
ByteMatrix = 128;
TVideoOut VideoOut("sample.avi",
ByteMatrix.Width(),
ByteMatrix.Height(),8);
for(int i=0; i<100;i++)
{
VideoOut << ByteMatrix;
}

```

Chương trình phần mềm thí nghiệm nổ bong bóng BUBBLE

```

#ifdef _TMS320C6X
#include "SDK/Bii/Bii.h"
using namespace Bii;
#define ENTRY_POINT main
#else
#include "SDK/Win32/IvApp/IvApp.h"
#define ENTRY_POINT IVApplication
#endif
// Khoi tao thu vien chua cac ham da duoc dinh nghia san trong Bi-I SDK
#include "C:\Program Files\InstantVisionISE\include\SDK\Bii\Bii.h"
#include "C:\Program Files\InstantVisionISE\include\InstantVision\BaseData\
TByteMatrix.h"
#include "C:\Program Files\InstantVisionISE\include\InstantVision\IO\TStd.h"
#include "C:\Program Files\InstantVisionISE\include\SDK\Bi\Ace16k\
TACE_IPL.h"
#include "C:\Program Files\InstantVisionISE\include\InstantVision\IO\
TVideoOut.h"

using namespace InstantVision;

```

```

int ENTRY_POINT(void)
{
TByteMatrix * GrayImage[250];
TStd Std(1);
TACE_IPL ace(IPL_MORPH);

TVideoOut VideoStream1;

/*Thiet lap cac thong so de bat giu anh, thoi gian bat giua anh la 0.03ms,
tuong ung voi so anh bat giu trong 1 giay la 33.000 anh
Gia tri nguong la 192*/
ace.Set_Int_Time(30); //Integration time = 0.03 millisecond
ace.Set_Precharge(192); //Precharge value = 192

Std.SetId(2);
Std << "Cat mau anh nhanh toc do 50 000 anh/giay";

// Khoi tao ma tran chua cac anh kich thuoc 128 x 128
// 250 anh duoc luu tru tam thoi trong bo nho cua DSP
for(int i = 0; i<250; i++)
{
    GrayImage[i] = new TByteMatrix(128,128);
}
/* Khoi tao file movie de luu tru anh sau moi lan xu ly,
kich thuoc cua moi frame anh la 128 x 128
so luong hinh trong mot giay la 25, phim chua cac anh muc xam */
VideoStream1.Open("D:\\Phong_CNN\\OpticalInputCapturing\\Data\\sample1.avi", 128,128,
8);
while(!GetStopStatus())
{
    for(int i = 0; i<250; i++)
    {
        if(GetStopStatus()) break;

        // Bat giu anh, dua anh den bo nho C_LAMI
        ace.CaptureOpt(C_LAMI);

        // Doc anh luu tru tu trong ACE16K vao bo nho DSP
        ace << C_LAMI >> *GrayImage[i];
    }
}

```

```

//Hien thi anh ket qua da luu tru vao trong bo nho DSP
for(int i = 0; i<250; i++)
{
    if(GetStopStatus()) break;
    // Dua ra man hinh may tinh chuoai anh thu duoc
    Std.SetId(1);
    Std << *GrayImage[i];
    // Dong thoi luu giu anh vao file movie tren may tinh
    VideoStream1 << *GrayImage[i];
}
}
for(int i = 0; i<250; i++)
    delete GrayImage[i];
VideoStream1.Close();
return 0;
}

```

Kết quả thí nghiệm

Chúng tôi đã thử nghiệm cho nổ bóng với rất nhiều lần khác nhau, kết quả xử lý được lưu giữ trong các file video để theo dõi quá trình nổ. Với tốc độ xử lý ảnh của camera thông thường hoặc bằng mắt ta không thể phát hiện ra các mảnh vỡ bắn ra từ quả bóng. Tuy nhiên khi sử dụng máy tính đa năng Bi-I v2 ta hoàn toàn có thể phát hiện ra các mảnh vỡ này. Hình 2.52 sau đây sẽ mô tả kết quả thử nghiệm:



Hình 2.52: Kết quả thử nghiệm nổ bóng lưu trên file AVI.

2. 4.2. HƯỚNG DẪN SỬ DỤNG CHƯƠNG TRÌNH THỬ NGHIỆM NỔ BÓNG

Mô hình thử nghiệm gồm hai phần chính: Hệ phát triển Bi-I có chương trình thử nghiệm nổ bong bóng **Bubble** và bong bóng đã được thổi căng. Phần mềm thử nghiệm nổ bong bóng **Bubble** đã được phát triển và nạp trong đĩa **CD “ các mô hình và chương trình thử nghiệm công nghệ xử lý ảnh tốc độ cao phục vụ cho nghiên cứu và đào tạo ”** của nhiệm vụ nghị định thư.

Địa chỉ của chương trình nguồn:

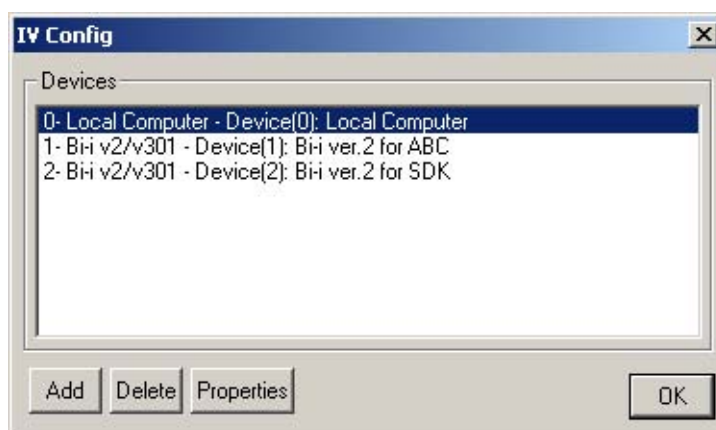
..\CNN-Projects\Bubble

Chạy ứng dụng Bubble:

Để chạy ứng dụng trên IVRun, cần phải thực hiện các bước sau:

- Thiết bị Bi-I cần đặt cấu hình trong hệ thống với IVConfig
- Số hiệu của thiết bị được đặt bởi IViConfig phải được lựa chọn trong IVRun trong hộp thoại Run Properties (danh mục Settings, mục Run)
- Nếu ứng dụng mặc định không đặt trong thiết bị thì ứng dụng cần được chỉ rõ (danh mục File, mục Open Application)

Thiết lập hộp thoại IVCONFIG



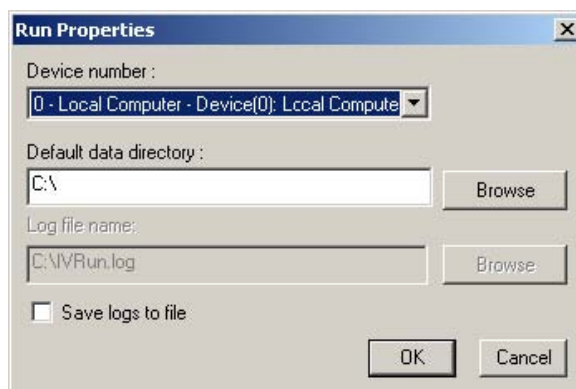
Hình 2.53: Hộp thoại IVConfig

Lựa chọn Devices:

2- Bi-i v2/v301 – Device(2): Bi-i ver.2 for SDK

Thiết lập hộp thoại Run Properties

Lựa chọn mục Run của Settings, gọi hộp thoại Run Properties. Hộp thoại như hình sau:



Hình 2.54: Hộp thoại Run properties

Trong mục Device Number, lựa chọn :

2- Bi-i v2/v301 – Device(2): Bi-i ver.2 for SDK

Trong mục Default data directory, lựa chọn đường dẫn tới thư mục lưu kết quả:

D:\MyCNN\Bubble\OpticalInputCapturing\Data

Thiết lập file .out chạy trên máy tính Bi-iv2:



Hình 2.55: Hộp thoại Properties-Local Computer

Trong mục Default Application, lựa chọn đường dẫn tới file.out. Ví dụ:

..\CD-Demo\Bubble\OpticalInputCapturing\Release_v2\Capture_v2.out

Thiết lập cấu hình Chương trình SDK

- Đặt các thông số sau trên nhãn 'Linker->Input' của hộp thoại 'Property Pages' cho tất cả các cấu hình:
- Thêm các thư viện: IvBase_msvc7_x86d.lib IvApp_msvc7_x86d.lib Utils_msvc7_x86.lib
- Tập định nghĩa module: .IVApp.def
- Vị trí của thư viện InstantVision và các tệp có thể đặt trong danh sách 'Tools/Options/Projects/VC++ Directories/'/'Include files' và 'Library files'.

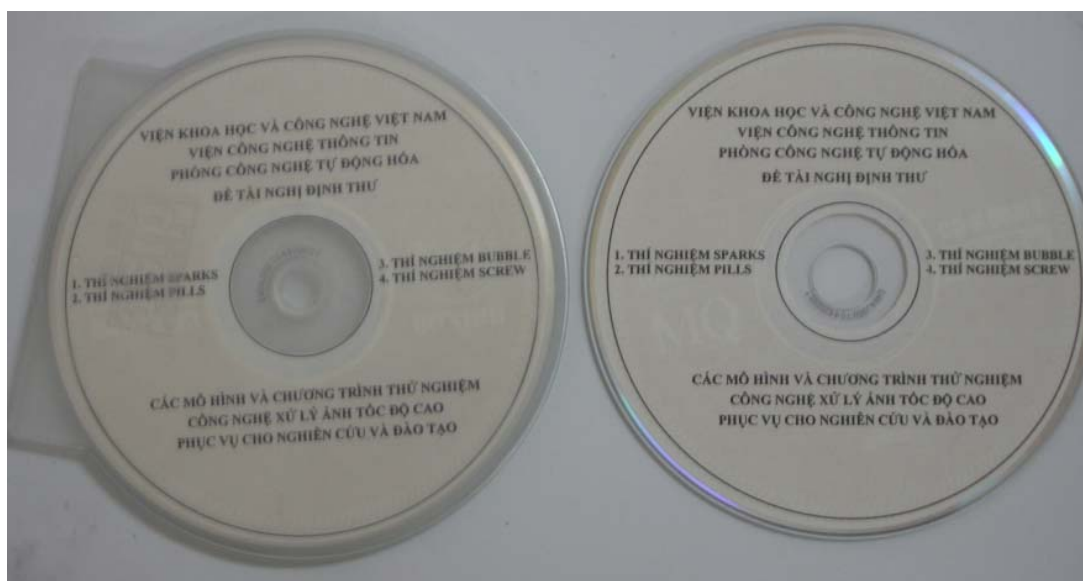
Sau khi thiết lập các cấu hình cần thiết nêu trên ta thực hiện các bước sau để thu được ảnh của bong bóng nổ:

1. Chỉnh tiêu cự của camera Bi-I hướng vào đối tượng bong bóng
2. Cho chương trình Bubble chạy
3. Kích cho bong bóng nổ
4. Kết quả quá trình nổ bong bóng được tự động ghi vào file video AVI của chương trình. Ta có thể xem lại dạng quay chậm trên màn hình.

2.5. Kết luận

Nhiệm vụ nghị định thư với sự hỗ trợ của Viện MTA SZTAKI Hungary đã nghiên cứu, làm chủ được công nghệ xử lý ảnh nhanh trên nền mạng nơ ron tế bào qua việc tiến hành xây dựng được các mô hình thí nghiệm bắt và xử lý ảnh tốc độ cao nêu trên. Việc phát hiện được sự xuất hiện của các sự kiện xảy ra nhanh (1-2 ms) như tia lửa điện (SPARKS), và nhận dạng được các sản phẩm viên thuốc (PILLS), ốc vít với tốc độ xử lý rất cao (>10000 fps) là các vấn đề khó nhất của công nghệ CNN mà nhiệm vụ nghị định thư đã thực hiện được. Các mô hình thí nghiệm SPARKS và PILLS của nhiệm vụ nghị định thư tương tự như của Viện MTA SZTAKI Hungary nhưng gọn nhẹ hơn. **Các kết quả này là các kết quả mới ở Việt nam, chưa có cơ sở nào ở Việt nam thực hiện được từ trước đến nay.** Đây cũng là các kết quả vượt trội của công nghệ xử lý ảnh nhanh CNN mà các hệ camera với các máy tính xử lý nối tiếp hiện hành chưa thực hiện được.

Bốn phần mềm CNN thí nghiệm phát hiện và phân tích tia lửa điện (SPARKS), quay chậm quá trình nổ bong bóng (BUBBLE), nhận dạng thuốc viên tốc độ cao (PILLS) và nhận dạng ốc vít tốc độ cao của nhiệm vụ nghị định thư đã được đúc trong đĩa **CD “ các mô hình và chương trình thử nghiệm công nghệ xử lý ảnh tốc độ cao phục vụ cho nghiên cứu và đào tạo** kèm theo các hướng dẫn mô tả thí nghiệm sẽ giúp ích cho việc đào tạo và nghiên cứu phát triển công nghệ CNN mới mẻ và đa triển vọng này.



3. KIẾN NGHỊ PHƯƠNG HƯỚNG PHÁT TRIỂN VÀ ỨNG DỤNG CÔNG NGHỆ CNN Ở VIỆT NAM

Mạng nơ ron tế bào CNN đã mở ra một hướng mới cho sự phát triển của khoa học tính toán. Chỉ với các chip CNN một lớp, khả năng tính toán và xử lý ảnh đã vượt xa các máy tính có hệ lệnh tuần tự hàng nghìn lần. Các chip CNN nhiều lớp thế hệ sau còn được tích hợp với các công nghệ mới như công nghệ cảm biến nano, quang tử sẽ được áp dụng để tạo ra các thiết bị siêu mạnh đáp ứng được các yêu cầu ngặt nghèo trong nhiều lĩnh vực của cuộc sống nói chung và trong công nghiệp nói riêng. Nhu cầu và tiềm năng ứng dụng công nghệ mạng nơ ron tế bào CNN phục vụ cho quốc phòng và an ninh là rất lớn, có thể góp phần quan trọng cho sự nghiệp củng cố quốc phòng và an ninh của đất nước.

Đây là một lĩnh vực khoa học công nghệ mới mẻ đầy triển vọng cho đa dạng ứng dụng. Với sự phát triển của công nghệ, các ứng dụng của CNN sẽ giải quyết nhiều vấn đề mà các hệ máy tính hiện hành chưa giải quyết được. Với các tính năng vượt trội, một ngày không xa công nghệ CNN sẽ thay thế các ứng dụng của các máy tính hiện hành và thâm nhập vào các hệ nhúng, hệ cơ điện tử tạo nên các sản phẩm và hệ thống thông minh có những chức năng xử lý tương tự như con người.

Tuy nhiên đây mới chỉ là tiềm năng, để biến công nghệ mới thành các phương tiện và thiết bị kĩ thuật với ứng dụng cụ thể cần tiếp tục đầu tư để nghiên cứu, thiết kế, chế tạo và thử nghiệm các sản phẩm đồng bộ thì mới có thể đạt được hiệu quả thiết thực

Để theo kịp xu thế phát triển và tạo ra các bước đột phá trong công nghệ cao, Việt nam cũng cần sớm đưa chương trình đào tạo, mở các môn học mới về tính toán nơ ron, công nghệ mạng nơ ron tế bào vào các trường đại học cho các bậc đào tạo đại học và sau đại học. Đặc biệt cần đẩy mạnh nghiên cứu về công nghệ CNN về cơ bản trong giai đoạn hiện nay và từng bước đầu tư trang thiết bị cho các nghiên cứu phát triển các ứng dụng chọn lọc trong thời gian tới.

Chúng tôi đã tổ chức Hội thảo với sự tham gia của các cán bộ khoa học ở các Viện, Trường và đã thảo luận về phương hướng phát triển và ứng dụng công nghệ CNN ở Việt nam.

Một số kiến nghị cụ thể cho giai đoạn trước mắt được liệt kê như sau:

a) Về đào tạo phát triển nguồn nhân lực

- Mở các khóa đào tạo, nâng cao trình độ cho các giảng viên đại học về công nghệ CNN và tính toán nơ ron ở trong nước và thực tập ở nước ngoài.
- Biên soạn các giáo trình giảng dạy về CNN và neuro-computing cho bậc đại học và cao học
- Từng bước đưa môn học về CNN và tính toán nơ ron vào chương trình đào tạo đại học và cao học như một số trường đại học tiên tiến trên thế giới (có thể mời giáo viên thỉnh giảng trong và ngoài nước)
- Mở các lớp chuyên đề và các đề tài làm đồ án tốt nghiệp ở các trường cho sinh viên

- Mở bộ môn tính toán nơ ron với phòng thí nghiệm công nghệ CNN ở một số trường đại học trọng điểm.
- Đào tạo các Nghiên cứu sinh làm các luận án tiến sỹ về công nghệ CNN
- Trao đổi giáo trình, phương pháp giảng dạy và các thí nghiệm thực hành về CNN với các cơ sở đào tạo trong khu vực và trên thế giới.

b) Về nghiên cứu cơ bản

Tiến hành các nghiên cứu cơ bản theo một số hướng sau:

- Nghiên cứu về tính ổn định và hỗn loạn (chaotic) của các mạng nơ ron phi tuyến tế bào nhiều lớp có trễ
- Nghiên cứu về các phương pháp giải hệ phương trình vi phân đạo hàm riêng (tuyến tính và phi tuyến) sử dụng mô hình mạng CNN
- Nghiên cứu các nguyên lý sóng đôi (twin-wave) trên CNN cho các ứng dụng bám mục tiêu di động
- Nghiên cứu mô hình mắt nhân tạo có khả năng nhận dạng vật thể thời gian thực
- Nghiên cứu các phương pháp mã hóa và giải mã thông tin trên dòng dữ liệu (spatial-temporal flows) như video, trường audio...
- Nghiên cứu các thuật toán CNN tính các đặc trưng sóng phi tuyến cho nhận dạng vật thể
- Nghiên cứu các phương pháp xử lý tín hiệu xúc giác tương tác cho các bài toán tinh vi như tay máy cầm các vật dễ vỡ, lật trang sách vv..
- Nghiên cứu các thuật toán CNN nhận dạng trường âm thanh tác động bất ngờ như tiếng nổ, tiếng động của các vụ đột kích vv...
- Nghiên cứu các thuật toán CNN nhận dạng nhanh cho chữ viết tay
- Nghiên cứu thuật toán CNN phối hợp (fusion) đa kênh, đa phổ cho định hướng địa hình đa mục tiêu
- Nghiên cứu các phương pháp xử lý nhanh ảnh y tế
- Nghiên cứu các mô hình CNN trong phỏng sinh học một số giác quan sinh vật như xúc giác, thính giác, vị giác, khứu giác, cảm nhận nhiệt độ của vùng da vv...
-

c) Biện pháp triển khai các hướng nghiên cứu cơ bản về CNN nêu trên

- Xây dựng các đề tài trong chương trình nghiên cứu cơ bản của nhà nước, đề tài cấp bộ và cấp trường, Viện
- Tuyển nghiên cứu sinh và học viên cao học triển khai các nghiên cứu làm luận án TS, Cao học trong các hướng CNN nêu trên
- Đề xuất các đề tài hợp tác quốc tế về CNN từ nguồn kinh phí nhà nước, từ quỹ khoa học Mỹ (NSF), chương trình hợp tác nghiên cứu Khoa học của

- Hợp tác nghiên cứu về CNN với các giáo sư ở các Lab CNN ở khu vực và thế giới như Nhật bản, Đài loan, Trung quốc, Mỹ, Hungary, Tây ban nha, Ý, Thổ nhĩ kỳ, Hà lan,
- Tổ chức các seminar về CNN ở các Viện, trường
- Tổ chức các hội thảo về CNN
- ...

d) Về nghiên cứu định hướng ứng dụng

Xây dựng các đề tài nghiên cứu theo một số hướng sau

- Nghiên cứu phát triển các hệ xử lý ảnh siêu âm, ảnh cắt lớp 2D, 3D thời gian thực
- Nghiên cứu phát triển hệ thống xử lý ảnh nhanh cho một số ứng dụng công nghiệp
- Nghiên cứu phát triển hệ thống nhận dạng đa mục tiêu di động cho các ứng dụng dân dụng và quốc phòng
- Nghiên cứu phát triển hệ thống xử lý dòng ảnh video theo công nghệ CNN cho các ứng dụng phục chế, nén, mã hóa phim ảnh vv..

e) Biện pháp triển khai các nghiên cứu CNN định hướng ứng dụng nêu trên

- Xây dựng các đề tài trong chương trình nghiên cứu trọng điểm về công nghệ thông tin, công nghệ tự động hóa của nhà nước, đề tài cấp bộ và cấp trường, Viện
- Hợp tác với các cơ sở sản xuất, dịch vụ, cơ sở y tế, các binh chủng có nhu cầu ứng dụng để cùng đầu tư nghiên cứu phát triển hệ thống
- Tuyển nghiên cứu sinh và học viên cao học triển khai các nghiên cứu làm luận án TS, Cao học trong các hướng nêu trên
- Đề xuất các đề tài hợp tác quốc tế về CNN từ nguồn kinh phí nhà nước, từ quỹ khoa học Mỹ (NSF), chương trình hợp tác nghiên cứu Khoa học của EU, quỹ hỗ trợ nghiên cứu châu Á của Canada, Hàn quốc, Thụy điển, APEC vv..
- Học hỏi và hợp tác nghiên cứu với các giáo sư ở các Lab CNN ở khu vực và thế giới như Nhật bản, Đài loan, Trung quốc, Mỹ, Hungary, Tây ban nha, Ý, Thổ nhĩ kỳ, Hà lan... đã triển khai các nghiên cứu liên quan
- ..

f) Các đề xuất khác

- Tăng cường quảng bá về công nghệ CNN và tính toán nơ ron để các cấp lãnh đạo và thế hệ trẻ hiện nay thấy được tiềm năng to lớn và các lợi ích quốc gia trong việc đầu tư cho phát triển lĩnh vực này.
- Thu thập các sách, bài báo tài liệu, thông tin, các công trình nghiên cứu, các luận tiến sỹ, cao học, các dự án và ứng dụng về CNN hàng năm để cung cấp thông tin cho các đề xuất nghiên cứu và đào tạo nêu trên (Có thể ghi lên CD sử dụng nội bộ)
- Xây dựng trang web chuyên đề về CNN và neuro-computing, tập hợp các thông tin và các đường link tới các phòng thí nghiệm, bộ môn liên quan trên thế giới
- Tăng cường công bố các công trình nghiên cứu và tổ chức hội nghị, hội thảo quốc gia và quốc tế liên quan đến công nghệ CNN và tính toán nơ ron
- ...

4. CÁC ÁN PHẨM ĐÃ CÔNG BỐ

- [1] **Trần Việt Phong, Phạm Thượng Cát**
Mạng nơ ron tế bào và ứng dụng
Tập chí Tự động hóa ngày nay Số: 65+66/2006 trang: 18-24
- [2] **Phạm Đức Long, Phạm Thượng Cát,**
"Công nghệ mạng nơ ron tế bào CNN và ứng dụng",
Tập chí Tin học và điều khiển học tập 22, số1-2006
- [3] **Phạm Thượng Cát**
Công nghệ mạng nơ ron tế bào CNN và khả năng ứng dụng trong các hệ cơ điện tử
Tuyển tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [4] **Trần Việt Phong, Phạm Thượng Cát**
Một số nghiên cứu về mô hình phỏng sinh học trong lĩnh vực thị giác nhân tạo
Tuyển tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [5] **Phạm Đức Long, Phạm Thượng Cát**
Ứng dụng công nghệ CNN (Cellular Neural Network) trong kiểm tra nhanh đường sắt.
Tuyển tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [6] **Bùi Thị Thanh Quyên, Phạm Thượng Cát**
Ứng dụng công nghệ mạng nơ ron tế bào (CNN) cho phân loại thuốc viên chuyên động tốc độ cao
Báo cáo tại Hội nghị "Kỷ niệm 30 năm ngày thành lập viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt Nam 1976-2006"
- [7] **Tài liệu Hội thảo lần thứ I về CNN**
"Công nghệ mạng nơ ron tế bào (CNN) và máy tính song song xử lý ảnh Bi-I
Hà nội 1/2006, 60 trang A4
- [8] **Tài liệu Hội thảo lần thứ II về CNN**
"Mạng nơ ron tế bào (CNN) và tính toán nơ ron
Hà nội 3/2007, 98 trang A4
- [9] **Tài liệu biên dịch**
"Thư viện xử lý ảnh INSTANT VISION cho siêu máy tính xử lý ảnh nhanh Bi-I
Hà nội 12, 2006, 122 trang A4

TÀI LIỆU THAM KHẢO

- [01] L. O.Chua and L. Yang, “Cellular neural networks: Theory,” *IEEE Trans. Circuits Syst., Vol. 35*, pp. 1257-1272, 1998.
- [02] L. O. Chua and Roska Tamas
Cellular Neural Networks and Visual Computing:
Foundations and Applications
Cambridge University Press 2002
- [03] Angela Slavova
Cellular Neural Networks: Dynamics and Modelling
Kluwer Academic Publishers 2003
- [04] L. O.Chua and L. Yang, “Cellular neural networks: Applications,” *IEEE Trans. Circuits Syst., Vol.35* pp. 1273-1290, 1998.
- [05] Ákos Zarándy, Rafael Domínguez-Castro & Servando Espejo, “Ultra High Frame Rate Focal Plane Image Sensor and Processor”, *IEEE Sensors Journal*, Volume 2, Issue 6, December 2002, Page(s): 559 – 565.
- [06] B. Mirzai, D. L’ým, and G. S. Moschytz, “Robust CNN templates: Theory and simulations,” in *Proc. IEEE Int. Workshop Cellular Neural Networks Applications*, Seville, Spain, June 1996
- [07] *Bi-i Programming Guide 1.1*, AnaLogic Computers Ltd, 2004
- [08] Chua L. O. and Roska T., *The CNN paradigm*, *IEEE Trans on Circuits and Systems I* 40 (1993) pp 147-156.
- [09] CNN Software Library for ACE4K chip (Templates and algorithms) Ver. 1.1, Analogical & Neural Computing laboratory Computer and Automation Institute Hungarian Academy of Sciences, 2000
- [10] D. Bálya, Cs. Rekeczky, T. Roska, "A Realistic Mammalian Retinal Model Implemented on Complex Tế bào CNN Universal Machine", 2002
- [11] Dávid Bálya, *Mammalian retina modeling and on-line learning on CNN architecture from wetware to silicon*, Ph.D. Dissertation.
- [12] Dániel Hillier, V. Binzberger, D. L. Vilarino, and Csaba Rekeczky, "Topographic Active Contour Techniques: Theory, Implementations and Comparisons", *Int. Journal on Circuit Theory and Applications*
- [13] E. Saatci, and V. Tavsanoğlu, *Fingerprint Image Enhancement using CNN Gabor-Type Filters*, *Proc. Seventh IEEE International Workshop on Cellular Neural Networks and their Application*, pp. 22-24, July 2002.
- [14] G. Tímár, and Cs. Rekeczky, “A Real-time Multi-target Tracking System with Robust Multi-channel CNN Algorithms”, *IEEE Trans. on Circuits and Systems*, Vol. 52, pp. 1358-1371, July, 2005
- [15] I. Szatmári, “Spatio-temporal Nonlinear Wave Metric for Binary and Gray-scale Object Comparison on Analogic Cellular Wave Computers”, *Int. Journal of Functional Differential Equations*, Vol.13, No.I, pp. 89-97, 2006
- [16] I. Petrás, M. Gilli “Complex dynamics in one-dimensional CNNs”, *Int. Journal of Circuit Theory and Applications*, Vol.34, No.1. pp. 3-20, 2006
- [17] I. Szatmári, Á. Zarándy, “High-speed Label Inspection System for Textile Industry”, *Proceedings of 10th IMEKO TC10 International Conference on Technical Diagnostics*, Budapest, Hungary, pp. 99-102, 2005

- [18] K. R. Crouse and L. O. Chua, *Methods for Image Processing in Cellular Neural Networks: A Tutorial*, IEEE Trans. on Circuits and Systems, Vol. 42, No. 10, pp. 583-601, October 1995.
- [19] Á. Zarándy, P. Földesy, P. Szolgay, Sz. Tőkés, Cs. Rekeczky, T. Roska, "Various implementations of topographic, sensory, cellular wave computers", *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'05)*, Kobe, Japan, Vol. 6, pp. 5802-5805, 2005
- [20] Leon O. Chua, Tamás Roska and Péter L. Venetianer
The CNN is Universal as the Turing Machine
IEEE Trans.on Circuits and Systems: Fundamental Theory and Applications Vol., 40, No.4, pp.289-291, 1993.
- [21] Analogic Computer Ltd. <http://www.analogic-computer.com>
- [22] Ngô Tứ Thành, *Ảnh vân tay đa cấp xám và thuật toán tự động xác định điểm đặc trưng dựa vào dòng chảy đường vân*, Tuyển tập các báo cáo khoa học hội nghị toàn quốc lần thứ 3 về tự động hóa, pp. 528-534, 04/1998
- [23] Phạm Đức Long, Phạm Thượng Cát, "*Công nghệ mạng nơ ron tế bào CNN và ứng dụng*", Tạp chí Tin học và điều khiển học tập 22, số 1-2006
- [24] Phạm Đức Long, Phạm Thượng Cát
Ứng dụng công nghệ CNN (Cellular Neural Network) trong kiểm tra nhanh đường sắt.
Báo cáo tại Hội nghị Cơ điện tử toàn quốc lần thứ 3, pp.363-371, Hà nội 10/2006.
- [25] Qun Gao, George S. Moschytz, *Fingerprint feature matching using CNNs*, IEEE ISCAS 2004, pp 73 – 76.
- [26] Tamás Roska
Cellular Wave Computers for Brain-like Spatial-Temporal Sensory Computing
IEEE Circuits and Systems Magazine, pp. 5-19, Second Quarter 2005.
- [27] Tamás Roska
Computational and Computer Complexity of Analogic Cellular Wave Computer.
Journal of Circuits, Systems and Computers Vol., 12, pp.539-562, 2003.
- [28] T. Roska, L.O.Chua, D. Wolf. T. Kozek, R.Tetzlaff and F.Duffer
Simulating Nonlinear Waves and Partial Differential Equations via CNN-Part I: Basic Techniques.
IEEE Trans.on Circuits and Systems: Fundamental Theory and Applications Vol., 42, No.10, pp.807-815, 1995.
- [29] T. Roska, L.O.Chua, D. Wolf. T. Kozek, R.Tetzlaff and F.Duffer
Simulating Nonlinear Waves and Partial Differential Equations via CNN-Part II: Typical Examples.
IEEE Trans.on Circuits and Systems: Fundamental Theory and Applications Vol., 42, No.10, pp.816-820, 1995.
- [30] Trần Việt Phong, Phạm Thượng Cát
Một số nghiên cứu về mô hình phỏng sinh học trong lĩnh vực thị giác nhân tạo
Báo cáo tại Hội nghị Cơ điện tử toàn quốc lần thứ 3, pp.168-172, Hà nội 10/2006.
- [31] Tamas Roska and L.O. Chua
The CNN Universal Machine: An Analogic Array Computer
IEEE Trans.on Circuits and Systems: Analog and Digital Signal Processing Vol., 40, No.3, pp.163-173, 1993.
- [32] Tamas Roska , L.O. Chua, T. Kozek and A. Zarandy
CNN Universal Chips Crank up the Computing Power

- IEEE Circuits and Devices: July 1996 pp.18-28
- [33] T. Kozek, T. Roska, and L. O. Chua, "Genetic algorithm for CNN template learning," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 392–402, June 1993.
 - [34] T. Roska, Á. Rodríguez-Vázquez. *Towards Visual Microprocessors.J.* Circuits, Systems, and Computers, 13, in print 2003.
 - [35] T. Roska and L.O.Chua. *The CNN Universal Machine: 10 years later.* Journal of Circuits, Systems, and Computers, Vol.12, No.4, August 2003
 - [36] T. Roska, "Cellular Wave Computers for Brain-Like Spatial-Temporal Sensory Computing", *Circuits and Systems Magazine*, second quarter 2005, pp. 5-19, 2005
 - [37] Z. Nagy, P. Szolgay, "Solving Partial Differential Equations on Emulated Digital CNN-UM Architectures", *Int. Journal of Functional Differential Equations*, Vol. 12
 - [38] Ákos Zarándy and Csaba Rekeczky
Bi-I: A Standalone Ultra High Speed Cellular Vision System.
IEEE Circuits and Systems Magazine pp 36-45, Second Quarter 2005.
 - [39] A.Wahab, S.H.Chin, E.C.Tan , *Novel approach to automated fingerprint recognition*, IEE Proc – Vis. Image Signal Process, Vol. 145, pp. 160-166, June 1998
 - [40] Ákos Zarándy, "The Art of CNN Template Design",
MTA SzTAKI Report , 1998
 - [41] Cellular Wave Computing Library (Templates, Algorithms, and Program) Ver.2.1
CSW-1-2007 MTA SzTAKI
 - [42] Eutecus Inc. Berkley <http://www.eutecus.com>
 - [43] Analogic and Neural Computing Lab
<http://lab.analogic.sztaki.hu/>

PHẦN PHỤ LỤC: TÒAN VĂN CÁC BÀI BÁO ĐÃ CÔNG BỐ

- [1] **Trần Việt Phong, Phạm Thượng Cát**
Mạng nơ ron tế bào và ứng dụng
Tạp chí Tự động hóa ngày nay Số: 65+66/2006 trang: 18-24
- [2] **Phạm Đức Long, Phạm Thượng Cát,**
"Công nghệ mạng nơ ron tế bào CNN và ứng dụng",
Tạp chí Tin học và điều khiển học tập 22, số1-2006
- [3] **Phạm Thượng Cát**
Công nghệ mạng nơ ron tế bào CNN và khả năng ứng dụng trong các hệ cơ điện tử
Tuyên tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [4] **Trần Việt Phong, Phạm Thượng Cát**
Một số nghiên cứu về mô hình phỏng sinh học trong lĩnh vực thị giác nhân tạo
Tuyên tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [5] **Phạm Đức Long, Phạm Thượng Cát**
Ứng dụng công nghệ CNN (Cellular Neural Network) trong kiểm tra nhanh đường sắt.
Tuyên tập Hội nghị Cơ điện tử toàn quốc lần thứ 3, Hà nội 10/2006.
- [6] **Bùi Thị Thanh Quyên, Phạm Thượng Cát**
Ứng dụng công nghệ mạng nơ ron tế bào (CNN) cho phân loại thuốc viên chuyển động tốc độ cao
Báo cáo tại Hội nghị "Kỷ niệm 30 năm ngày thành lập viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt Nam 1976-2006"

**VIỆN KHOA HỌC VÀ CÔNG NGHỆ VIỆT NAM
VIỆN CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO TÓM TẮT
ĐỀ TÀI NGHỊ ĐỊNH THƯ**

**HỢP TÁC NGHIÊN CỨU PHÁT TRIỂN
CÁC HỆ THỐNG XỬ LÝ ẢNH NHANH TRÊN CƠ SỞ
ÁP DỤNG CÔNG NGHỆ MẠNG NƠON
PHI TUYẾN TẾ BÀO**

Chủ nhiệm đề tài: PGS. TSKH. PHẠM THƯỢNG CÁT

HÀ NỘI - 2007

BÁO CÁO ĐÁNH GIÁ TÓM TẮT NHIỆM VỤ NGHỊ ĐỊNH THƯ

I. THÔNG TIN TỔNG QUAN:

1. **TÊN NHIỆM VỤ:** “ *Hợp tác nghiên cứu phát triển các hệ thống xử lý ảnh nhanh trên cơ sở áp dụng công nghệ mạng nơron phi tuyến tế bào*”

2. **THỜI GIAN THỰC HIỆN:**

Bắt đầu: 1/01/2005

Kết thúc: 31/12/2006

3. **ĐỐI TÁC VIỆT NAM:**

a. *Tên cơ quan chủ trì Việt Nam (tên, địa chỉ, website):*

Tên: Viện Công nghệ thông Tin thuộc Viện Khoa học Việt nam

Địa chỉ: 18 Hoàng Quốc Việt, Hà nội

Web site: <http://www.ioit.ac.vn>

b. *Chủ nhiệm đề tài*

Tên: **PGS. TSKH Phạm Thượng Cát**

Điện thoại cơ quan: 04-8361445

Fax: 04-8363485

Email: ptcat@ioit.ncst.ac.vn

Điện thoại di động: 0913307703

c. *Danh sách 05 cán bộ khác trực tiếp tham gia nghiên cứu*

(tên, điện thoại cơ quan, fax, email, điện thoại di động)

TS. Phạm Minh Tuấn, Viện Công nghệ Thông Tin, 04-8363485

Th.S. Trần Việt Phong, Viện Công nghệ Thông Tin, 04-8363484

Th.S. Bùi Thị Thanh Quyên, Viện Công nghệ Thông Tin, 04-8363484

Th.S. Phạm Ngọc Minh, Viện Công nghệ Thông Tin, 04-8363484

Th.S. Phạm Đức Long, Đại Học Thái Nguyên, 0912551589

GS. TSKH Nguyễn Đức Cường, Viện Tên Lửa, 069516064

4. **ĐỐI TÁC NƯỚC NGOÀI:**

a. *Tên Cơ quan đối tác nghiên cứu nước ngoài (tên, địa chỉ, website)*

Tên: Viện nghiên cứu máy tính và tự động hóa thuộc Viện Hàn lâm khoa học Hungary (**MTA SzTAKI**)

Magyar Tudományos Akademia Szamitás Technikai es Automatizalási Kutató Intezet)

Địa chỉ: 1111 Budapest Kende ut. 13-17, Hungary

Web site: <http://www.sztaki.hu>

b. *Chủ nhiệm đề tài (tên, điện thoại cơ quan, fax, email, điện thoại di động)*

Tên: **Viện sỹ Roska Tamas**

Điện thoại cơ quan: +36 1 279 6151, +36 1 279 6155, +36 1 209 5263

Fax: +36 1 209 5264

Email: roska@sztaki.hu

c. *Danh sách 05 cán bộ khác trực tiếp tham gia*

Prof. Szolgai Peter, MTA SzTAKI, szolgai@sztaki.hu, +36 1 279 6128,

Dr. Rekeczky Csaba, MTA SzTAKI, rcsaba@lutra.sztaki.hu, +36 1 279 6131,

Dr. Zarandy Akos, MTA SzTAKI, zarandy@sztaki.hu, +36 1 279 6131

Dr. Barfai Gusztav, MTA SzTAKI, guztav.barfai@sztaki.hu, +36 1 371 1660

Prof. Radvanyi Andras, MTA SzTAKI, radvanyi@sztaki.hu +36 1 279 6128

5. **KINH PHÍ PHÍA VIỆT NAM:**

a. Tổng kinh phí:

- Kinh phí được hỗ trợ từ ngân sách Nhà nước: 800 000 000 VND

b. Kinh phí đã chi 800 000 000 VND

6. **KINH PHÍ CỦA ĐỐI TÁC (ƯỚC TÍNH) 200 000 Euro**

II. KẾT QUẢ VÀ ĐÁNH GIÁ:

1. **CÁC NỘI DUNG CHÍNH ĐÃ TRIỂN KHAI CỦA PHÍA VIỆT NAM:**

- a. Nghiên cứu cơ bản về các phương pháp tính toán và xử lý ảnh sử dụng mạng nơron tế bào CNN.
- b. Nghiên cứu làm chủ công nghệ và hệ thống phần mềm phát triển xử lý ảnh nhanh Bi-i 10.000 ảnh/s.
- c. Nghiên cứu xây dựng 02 mô hình nhận dạng và xử lý ảnh nhanh sử dụng công nghệ mạng nơron tế bào CNN phục vụ cho nghiên cứu và đào tạo bao gồm:
 - Mô hình thu ảnh tia lửa điện tốc độ > 10 000 fps
 - Mô hình nhận dạng xử lý ảnh phân loại sản phẩm (thuốc viên, ốc vít...) tốc độ cao
- d. Phát triển các chương trình phần mềm công cụ và phần mềm ứng dụng trên nền công nghệ mạng nơron tế bào CNN.
- e. Tổ chức 02 seminar, hội thảo khoa học về công nghệ CNN và Bi-i với sự hỗ trợ của các chuyên gia Hungary.
- f. Hợp tác với Hungary trong đào tạo công nghệ xử lý ảnh nhanh cho các cán bộ Việt Nam.
- g. Đề xuất và kiến nghị về phương pháp phát triển và ứng dụng công nghệ xử lý ảnh nhanh CNN cho công nghiệp và quốc phòng.

2. CÁC NỘI DUNG HỢP TÁC CHÍNH VỚI ĐỐI TÁC:

a. Thu thập, trao đổi thông tin về các kết quả nghiên cứu về công nghệ CNN đã được triển khai tại Hungary

b. Triển khai các đợt khảo sát, nghiên cứu học hỏi và trao đổi học thuật về công nghệ xử lý ảnh nhanh tại các cơ sở nghiên cứu và đào tạo tại Hungary bao gồm

- Phòng thí nghiệm tính toán nơ ron và tương tự - số (Analogical and Neural Computing Systems Laboratory) thuộc Viện MTA SzTAKI

- Khoa Công nghệ Thông tin Đại học Veszprem Hungary

- Khoa Công nghệ Thông tin Đại học Pázmány Budapest Hungary

- Doanh nghiệp Khoa học công nghệ Analogic Computer Ltd. Của MTA SzTAKI

c. Mở khóa đào tạo cho các cán bộ của Việt nam về công nghệ lập trình cho mạng CNN và thực hành trên các siêu máy tính xử lý ảnh Bi-I tại Hungary

d. Tiếp đón 2 đoàn của Viện MTA SzTAKI gồm Viện sỹ Keviczky László và Viện sỹ Róska Tamás sang Việt nam vào tháng 10 và tháng 12 năm 2007. Các đoàn Hungary đã trình bày các phương hướng và kết quả nghiên cứu mới nhất về CNN và tham quan khảo sát các phòng thí nghiệm của Viện CNTT, các khoa CNTT và Khoa Điện của các trường đại học Bách khoa Hà nội, Đại học quốc gia Hà nội và Đại học Thái nguyên. Hai bên đã thảo luận và nhất trí tiếp tục hợp tác phát triển đào tạo và nghiên cứu về lĩnh vực CNN ở Việt nam.

3. CÁC KẾT QUẢ ĐẠT ĐƯỢC:

a. Các nghiên cứu cơ bản về CNN bao gồm

- Cấu trúc, mô tả toán học, tính ổn định của hệ động lực mạng nơ ron tế bào
- Các phương pháp thiết kế các ma trận trọng liên kết của mạng CNN
- Mối quan hệ giữa CNN với các hệ phương trình vi phân đạo hàm riêng tuyến tính và phi tuyến
- Các template (A, B,z) của CNN trong xử lý ảnh nhanh
- Cấu trúc của máy tính tế bào vạn năng xử lý luồng dữ liệu CNN UMF (Cellular Neural Network Universal Machin over Flows)

b. Nghiên cứu và làm chủ được công nghệ mạng nơ ron tế bào CNN, hệ thống phần cứng và hệ phát triển phần mềm cho máy tính xử lý ảnh nhanh Bi-I

c. Xây dựng 02 mô hình thu nhận và xử lý ảnh tốc độ cao > 10000 fps cho nghiên cứu và đào tạo

d. Tổ chức 03 Hội thảo khoa học về công nghệ CNN cho đông đảo cán bộ khoa học tại các Viện nghiên cứu, Trường đại học và một số công ty công nghệ cao

- Hội thảo về CNN lần thứ nhất được tổ chức vào tháng 1/2006 tại Hà nội.

- Hội thảo về CNN lần thứ hai được tổ chức vào tháng 3/2007 tại Hà nội.

- Hội thảo về CNN lần thứ hai được tổ chức vào tháng 12/2007 tại Đại Học Thái Nguyên.

- e. Đăng tải 6 công trình về CNN trong các tạp chí, hội nghị khoa học trong nước
- f. Nâng cao năng lực của cán bộ khoa học và công nghệ Việt nam:
 - 05 cán bộ được đào tạo và học hỏi về công nghệ CNN tại các phòng thí nghiệm, nghiên cứu phát triển về CNN tại Hungary
 - 01 thạc sỹ đã bảo vệ thành công luận án về CNN
 - 04 NCS đang làm luận án tiến sỹ về CNN
- g. Xây dựng được 01 phòng thí nghiệm nhỏ về công nghệ CNN tại Viện Công nghệ thông tin
- h. Hình thành được một số nhóm nghiên cứu về CNN ở
 - Viện Công nghệ thông tin thuộc Viện KH và CN Việt nam
 - Viện Công nghệ thông tin thuộc Trung tâm khoa học kỹ thuật và công nghệ Quân sự
 - Khoa Công nghệ thông tin Đại học Thái nguyên
- i. Các kết quả khác
 - Bước đầu đưa được hướng công nghệ CNN và tính toán mờ vào Việt nam.
 - Tiếp tục duy trì và phát triển được quan hệ hợp tác KH&CN với các Viện sỹ Viện Hàn lâm khoa học Hungary và các giáo sư đầu ngành làm việc tại Viện nghiên cứu Máy tính và Tự động hóa MTA SzTAKI Hungary

4. ĐÁNH GIÁ:

- a. Đánh giá về chất lượng kết quả KH&CN của Nhiệm vụ (có thể so sánh với một số nội dung nghiên cứu có liên quan được tiến hành trong nước từ trước đến nay).
 - Lĩnh vực công nghệ mạng mờ tế bào là lĩnh vực mới, chưa được tiến hành nghiên cứu từ trước đến nay ở trong nước. Đề tài đã hoàn thành đầy đủ các nội dung nhiệm vụ đặt ra với chất lượng tốt nhất.
- b. Ứng dụng vào thực tiễn đời sống (sản xuất, kinh doanh).
 - Công nghệ CNN là công nghệ cao mới nổi trội có nhiều tiềm năng ứng dụng lớn. Tuy nhiên hiện nay trên thế giới sự phát triển của CNN vẫn đang trong giai đoạn bùng nổ về nghiên cứu cơ bản, phát triển các giải pháp đột phá và ứng dụng thử nghiệm là chính. Việc ứng dụng CNN vào thực tế hiện đang tập trung vào các đơn đặt hàng của NASA và Cộng Hòa liên bang Đức do giá thành của các chip CNN còn cao. Việc ứng dụng vào sản xuất kinh doanh đại trà còn rất khiêm tốn và được dự báo vào thời gian tới khi các chip CNN được sản xuất hàng loạt với giá thành rẻ. Năm bắt xu thế này đề tài đã tiến hành các nghiên cứu về khả năng ứng dụng CNN vào công nghiệp và an ninh quốc phòng
- c. Đánh giá về năng lực và tính tiên tiến (đi trước, kinh nghiệm) của đối tác nước ngoài.

Viện sỹ Roska Tamás là một trong 2 nhà khoa học phát minh ra máy tính vạn năng tế bào CNN trên thế giới. Phòng thí nghiệm về CNN của Viện sỹ Roska Tamás tại Viện MTA SzTAKI là một phòng thí nghiệm hàng đầu về CNN trên thế giới với đông đảo các cán bộ nghiên cứu.

d. Nêu và đánh giá về tầm quan trọng của vai trò hỗ trợ của đối tác nước ngoài (rút ngắn thời gian nghiên cứu trong nước, kết quả thu được có chất lượng tương đương quốc tế, ...).

Đề tài may mắn được hợp tác trực tiếp với Viện sỹ Roska Tamás và nhóm nghiên cứu của Ông tại Viện MTA SzTAKI và đã nhận được nhiều hỗ trợ quý giá về tài liệu, phần mềm, về đào tạo, tham quan các thí nghiệm trực tiếp và thường xuyên được trao đổi, khảo học nhanh chóng qua e-mail và Internet. Sự hợp tác giúp đỡ nhiệt tình này đã rút ngắn được nhiều thời gian nghiên cứu trong nước. Khi tổ chức các hội thảo về CNN, đề tài được Viện sỹ Roska Tamás tư vấn về nội dung và hỗ trợ các tài liệu cần thiết và gửi bài cho Hội thảo về các kết quả nghiên cứu mới nhất trong lĩnh vực CNN.

III. ĐỀ XUẤT KIẾN NGHỊ:

1. KIẾN NGHỊ

Để theo kịp xu thế phát triển và tạo ra các bước đột phá trong công nghệ cao, Việt nam cũng cần sớm đưa chương trình đào tạo, mở các môn học mới về tính toán nơ ron, công nghệ mạng nơ ron tế bào vào các trường đại học cho các bậc đào tạo đại học và sau đại học. Đặc biệt cần đẩy mạnh nghiên cứu về công nghệ CNN về cơ bản trong giai đoạn hiện nay và từng bước đầu tư trang thiết bị cho các nghiên cứu phát triển các ứng dụng chọn lọc trong thời gian tới. Các kiến nghị chi tiết đề tài đã tổng hợp trong báo cáo ” ***Kiến nghị về phương hướng phát triển công nghệ CNN và tính toán nơ ron ở Việt nam***”.

Một số phương hướng chính như sau

a. Về đào tạo phát triển nguồn nhân lực

- Mở các khóa đào tạo, nâng cao trình độ cho các giảng viên đại học về công nghệ CNN và tính toán nơ ron ở trong nước và thực tập ở nước ngoài.
- Biên soạn các giáo trình giảng dạy về CNN và neuro-computing cho bậc đại học và cao học
- Từng bước đưa môn học về CNN và tính toán nơ ron vào chương trình đào tạo đại học và cao học như một số trường đại học tiên tiến trên thế giới (có thể mời giáo viên thỉnh giảng trong và ngoài nước)
- Mở bộ môn tính toán nơ ron với phòng thí nghiệm công nghệ CNN ở một số trường đại học trọng điểm.
- Đào tạo các Nghiên cứu sinh làm các luận án tiến sỹ về công nghệ CNN
- Trao đổi giáo trình, phương pháp giảng dạy và các thí nghiệm thực hành về CNN với các cơ sở đào tạo trong khu vực và trên thế giới.

b) Về nghiên cứu cơ bản

- Nghiên cứu về tính ổn định và hỗn loạn (chaotic) của các mạng nơ ron phi tuyến tế bào nhiều lớp có trễ
- Nghiên cứu về các phương pháp giải hệ phương trình vi phân đạo hàm riêng (tuyến tính và phi tuyến) sử dụng mô hình mạng CNN
- Nghiên cứu các nguyên lý sóng, các thuật toán CNN tính các đặc trưng sóng phi tuyến trên CNN cho các ứng dụng xử lý ảnh và nhận dạng phân loại bám mục tiêu di động
- Nghiên cứu các phương pháp mã hóa và giải mã thông tin trên dòng dữ liệu (spatial-temporal flows) như video, trường audio...
- Nghiên cứu các thuật toán CNN nhận dạng nhanh cho chữ viết tay

- Nghiên cứu thuật toán CNN phối hợp (fusion) đa kênh, đa phổ cho định hướng địa hình đa mục tiêu
 - Nghiên cứu các phương pháp xử lý nhanh ảnh y tế
 - Nghiên cứu các mô hình CNN trong phòng sinh học một số giác quan sinh vật như thị giác, xúc giác, thính giác, vị giác, khứu giác, cảm nhận nhiệt độ của vùng da vv...
- c) Về nghiên cứu định hướng ứng dụng
- Xây dựng các đề tài nghiên cứu theo một số hướng sau
- Nghiên cứu phát triển các hệ xử lý ảnh siêu âm, ảnh cắt lớp 2D, 3D thời gian thực
 - Nghiên cứu phát triển hệ thống xử lý ảnh nhanh cho một số ứng dụng công nghiệp
 - Nghiên cứu phát triển hệ thống nhận dạng đa mục tiêu di động cho các ứng dụng dân dụng và quốc phòng
 - Nghiên cứu phát triển hệ thống xử lý dòng ảnh video theo công nghệ CNN cho các ứng dụng phục chế, nén, mã hóa phim ảnh vv..
- d) Biện pháp triển khai các hướng nghiên cứu cơ bản và triển khai về CNN nêu trên
- Xây dựng các đề tài trong chương trình nghiên cứu cơ bản của nhà nước, đề tài cấp bộ và cấp trường, Viện
 - Hợp tác nghiên cứu cơ bản về CNN với các giáo sư ở các Lab CNN ở khu vực và thế giới như Nhật bản, Đài loan, Trung quốc, Mỹ, Hungary, Tây ban nha, Ý, Thổ nhĩ kỳ, Hà lan,
 - Xây dựng các đề tài trong chương trình nghiên cứu trọng điểm về công nghệ thông tin, công nghệ tự động hóa của nhà nước, đề tài cấp bộ và cấp trường, Viện
 - Hợp tác với các cơ sở sản xuất, dịch vụ, cơ sở y tế, các bệnh chủng có nhu cầu ứng dụng để cùng đầu tư nghiên cứu phát triển hệ thống
 - Tuyển nghiên cứu sinh và học viên cao học triển khai các nghiên cứu làm luận án TS, Cao học trong các hướng nêu trên
 - Đề xuất các đề tài hợp tác quốc tế về CNN từ nguồn kinh phí nhà nước, từ quỹ khoa học Mỹ (NSF), chương trình hợp tác nghiên cứu Khoa học của EU, quỹ hỗ trợ nghiên cứu châu Á của Canada, Hàn quốc, Thụy điển, APEC vv..
 - Học hỏi và hợp tác nghiên cứu với các giáo sư ở các Lab CNN ở khu vực và thế giới như Nhật bản, Đài loan, Trung quốc, Mỹ, Hungary, Tây ban nha, Ý, Thổ nhĩ kỳ, Hà lan... đã triển khai các nghiên cứu liên quan
- e) Các đề xuất khác
- Tăng cường quảng bá về công nghệ CNN và tính toán nơ ron để các cấp lãnh đạo và thế hệ trẻ hiện nay thấy được tiềm năng to lớn và các lợi ích quốc gia trong việc đầu tư cho phát triển lĩnh vực này.
 - Tăng cường công bố các công trình nghiên cứu và tổ chức hội nghị, hội thảo quốc gia và quốc tế liên quan đến công nghệ CNN và tính toán nơ ron

2. DỰ KIẾN CÁC NỘI DUNG SẼ TRIỂN KHAI TIẾP THEO (trên cơ sở kế thừa các kết quả của Nhiệm vụ, về triển khai trong nước cũng như với đối tác nước ngoài)

- a. Duy trì hợp tác với Viện sỹ Roska Tamás và Viện MTA SzTAKI trong đào tạo, nghiên cứu và ứng dụng công nghệ CNN
- b. Tiếp tục triển khai các nghiên cứu cơ bản về CNN định hướng như sau:
 - Nghiên cứu các template của mạng CNN và phương pháp thiết kế các template mới các thuật toán CNN trong xử lý ảnh tốc độ cao.

- Nghiên cứu các cấu trúc và thuật toán CNN mới cho mô phỏng thời gian thực các hệ thống phức tạp được mô hình bằng các phương trình vi phân đạo hàm riêng.
 - Nghiên cứu các đặc trưng sóng phi tuyến lan truyền trên mạng CNN và áp dụng cho bài toán nhận dạng ảnh động
 - Nghiên cứu về tính ổn định và hỗn loạn (chaotic) của hệ động lực phi tuyến của mạng nơ ron tế bào CNN.
- c. Triển khai viết giáo trình bậc đại học và sau đại học về CNN và tính toán nơ ron cho các trường Đại học ở Việt nam.
- d. Xây dựng các đề tài cho sinh viên làm đề án tốt nghiệp cao học và đại học về CNN
- e. Tiếp tục thu hút các nghiên cứu sinh trẻ làm luận án tiến sỹ về CNN

Hà nội ngày 30 tháng 12 năm 2007
Chủ trì nhiệm vụ

PGS. TSKH Phạm Thượng Cát

Phiếu đăng ký kết quả thực hiện nhiệm vụ khoa học và công nghệ
(Ban hành kèm theo Quyết định số 03/2007/QĐ-BKH-CN ngày 16/3/2007
của Bộ trưởng Bộ Khoa học và Công nghệ)

<p>1. Tên đề tài:</p> <p>“Hợp tác nghiên cứu phát triển các hệ thống xử lý ảnh nhanh trên cơ sở sử dụng công nghệ mạng nơ ron phi tuyến tế bào”</p> <p>2. Cấp đề tài: Nhà nước</p> <p>3. Mã số đề tài : 29/2005/HĐ-NĐT....Thuộc Chương trình:....Nghị định thư với Hungary</p>
<p>4. Cơ quan chủ trì đề tài: Viện Công nghệ Thông Tin</p> <p><i>Địa chỉ: 18 Hoàng Quốc Việt Hà nội</i></p> <p><i>Điện thoại: 7564405</i></p>
<p>5. Cơ quan chủ quản: Viện Khoa Học và Công nghệ Việt nam</p> <p><i>Địa chỉ: 18 Hoàng Quốc Việt Hà nội</i></p> <p><i>Điện thoại: 7567305</i></p>
<p>6. Tổng kinh phí (triệu đồng) : 1600</p> <p><i>Trong đó, từ ngân sách Nhà nước (triệu đồng): 800</i></p>
<p>7. Thời gian thực hiện:24....tháng, bắt đầu từ tháng....01../.2005.....</p> <p style="text-align: right;">kết thúc....12...../.....2006.....</p>
<p>8. Chủ nhiệm đề tài : Họ và tên Phạm Thượng Cát</p> <p style="text-align: center;">Học hàm, học vị: PGS. TSKH</p>
<p>9. Danh sách cá nhân tham gia nghiên cứu (ghi họ tên, học hàm, học vị):</p> <ol style="list-style-type: none">1. PGS. TSKH Phạm Thượng Cát2. TS Phạm Minh Tuấn3. ThS. Phạm Đức Long4. Th.S. Trần Việt Phong5. ThS. Bùi Thị Thanh Quyên6. KS. Phan Minh Tân

- 7. ThS. Phạm Hồng Long
- 8. ThS. Nguyễn Tuấn Minh
- 9. ThS. Vũ Đức Thái
- 10. ThS. Phạm Ngọc Minh
- 11. GS. TSKH Nguyễn Đức Cường

10. Đề tài được nghiệm thu chính thức theo Quyết định số....296/QĐ-BKHHCN.....
ngày...10.....tháng..12.....năm...2007.....của.. Bộ Khoa Học và Công nghệ.....

11. Hợp nghiệm thu chính thức ngày. 22..tháng12.năm..2007. tại. Viện Công Nghệ Thông Tin

12. Bảo mật thông tin: *A - Không mật*

13. Sản phẩm giao nộp (ghi số lượng cụ thể):

- 13.1. Phiếu đăng ký....01....bản.
- 13.2. Biên bản họp nghiệm thu: 01 bản sao.
- 13.3. Báo cáo tổng kết....01.....quyển.
- 13.4. Báo cáo tóm tắt...01quyển.
- 13.5. Đĩa CD...01.....đĩa

14. Chủ nhiệm đề tài
(Ghi rõ họ tên, học hàm học vị và ký)

PGS. TSKH Phạm Thượng Cát

15. Xác nhận của cơ quan chủ trì
(*Thủ trưởng ký tên, đóng dấu*)