

XỬ LÝ ẢNH Y TẾ 4D - CT CHỊU NÉN SỬ DỤNG MẠNG NƠON TẾ BÀO

CÁP THANH TÙNG ¹
PHẠM THƯỢNG CÁT ²

¹Phòng CNTT-TV, Trường Đại học Sư Phạm Thái Nguyên

²Viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt Nam.

Abstract. With regards of 3D image, we use the concept of voxel (Volumetric cell) instead of pixel (picture cell). A single voxel consists of (x, y, z) for 3 space dimensions. Database of voxels can describe large structures and can be applied in many fields: architecture, video game, geology, astronomy, satellite image ... and, especially, in processing medical image. It is noted in the research of anatomy image from computed tomography of brain, lung, heart, thorax, hips... that those structures continuously and periodically change their position and size (due to respiration rhythm, pulse break, movement of muscles and articulations...) so the presence of time factor in parameter set of voxel is required. The processing of motion image concerns 4D-CT (4 Dimension Computed Tomography) scanning techniques and requires IR (Image Registration) techniques. There are many image registration approaches corresponding to various registration criteria (optical flow speed, calculation model, geometric feature ...). They are widely applied in analysis and calculation of medical images. Nevertheless, each approach still has issues to be addressed (both accuracy and calculation speed optimization). With medical image in general and with analysis of lung polyps diagnosis image, there is always considerable discrepancy. In general, the result will be rather good if intensity in image is unchanged, but this is impractical because lung is always in motion (due to respiration rhythm) and intensity on object (lung tissue) will be changed accordingly, which in turn leads to many errors in analysis. We call CT image of elastic tissues compressible image. Moreover, we investigate the calculation of optical flow speed of 4D-CT compressible image. This processing has a large number of image data and the calculation of optical flow speed is very complicated. So high processing speed is required and this is a great challenge for the PC. In this article, in order to improve calculation speed, we propose a processing with CNN for calculating 4D optical flow speed in real time mode.

Tóm tắt. Với ảnh 3 chiều, ta sử dụng khái niệm voxel (Volumetric Cell) thay vì pixel (Picture Cell) của 2D. Mỗi phần tử voxel đơn chứa đựng các tham số (x, y, z) cho 3 chiều không gian. Tập hợp dữ liệu của các voxel có thể mô tả cho các hình thể lớn, ứng dụng trong nhiều lĩnh vực: kiến trúc, hoạt hình (Video Game), địa chất, thiên văn, ảnh vệ tinh,... và đặc biệt trong xử lý ảnh y tế. Khi nghiên cứu các ảnh phẫu thuật có được từ các ảnh chụp cắt lớp CT (Computed Tomography): não, phổi, tim, lồng ngực, xương chậu... các khối hình này còn liên tục thay đổi vị trí và kích thước theo chu kỳ (nhịp thở, mạch đập, sự vận động cơ, khớp...) nên cần tới yếu tố thời gian (t) trong bộ tham số của voxel. Do đó việc xử lý ảnh động liên quan tới kỹ thuật scan 4D-CT (4 Dimension Computed Tomography). Quá trình xử lý ảnh 4D-CT cần đến kỹ thuật xác nhận ảnh IR (Image Registration) mà bản chất là xác định các điểm ảnh tương thích giữa hai ảnh liên kề trong quá trình chuyển động. Có nhiều phương pháp xác nhận ảnh khác nhau, ứng với các tiêu chí nhận dạng khác nhau (theo tốc độ của luồng ảnh (Optical Flow), theo mô hình tính toán, theo đặc điểm hình học...). Chúng

được ứng dụng rộng rãi trong các phân tích và tính toán cho dòng ảnh y tế. Tuy vậy còn khá nhiều tồn tại trong mỗi phương pháp, cần được giải quyết (cả về độ chính xác lẫn việc tối ưu hoá tốc độ tính toán). Với các ảnh y học nói chung và với việc phân tích ảnh cho chẩn đoán u phổi nói riêng, luôn vấp phải các sai số lớn do tính co giãn của các tế bào và các nhiễu từ ảnh thu được. Nhìn chung các kết quả sẽ khá tốt nếu cường độ ánh sáng trên mỗi điểm ảnh không thay đổi (phương pháp của Horn/Schunck), nhưng đây là điều không thực tế vì phổi luôn vận động theo nhịp thở và vì vậy cường độ sáng trên vật thể (mô phổi) cũng biến đổi theo, gây ra nhiều sai sót cho quá trình phân tích. Ta gọi các ảnh CT của các mô chịu tác động co giãn là các ảnh chịu nén (Compressible Image). Ở đây, ta khảo sát việc tính toán tốc độ dòng ảnh (Optical Flow) của ảnh chịu nén CT 4D. Các ảnh 4D có khối lượng điểm ảnh lớn với các tính toán dòng ảnh có độ phức tạp của tính toán rất lớn đòi hỏi tốc độ xử lý cao là một thách thức lớn cho các máy PC xử lý tuần tự hiện nay. Để nâng cao tốc độ tính toán, tác giả đã đề xuất một mô hình mạng nơron tế bào xử lý song song có khả năng tính tốc độ luồng ảnh 4D trong chế độ thời gian thực.

1. ĐẶT VẤN ĐỀ

Từ những năm 1985, kỹ thuật xử lý luồng ảnh 2 chiều đã được đề xuất để thu nhận dữ liệu vận tốc của mỗi điểm ảnh chuyển động. Ngày nay, với không gian 3 chiều các quá trình xử lý trở nên phức tạp hơn với đơn vị xử lý là các voxel chứ không còn là pixel nữa. Các ảnh chụp cắt lớp 4 chiều (4D-CT) được ứng dụng nhiều trong y tế, nó trợ giúp đắc lực cho các chẩn đoán y học cũng như quá trình đánh giá hoạt động của u bướu được chính xác hơn, đặc biệt ở khu vực lồng ngực (u phổi). Các ảnh 4D-CT có thể chứa tới 20 khối ảnh (Image Volume) 3 chiều cho mỗi pha, với kích thước đạt tới $512 \times 512 \times 200$ voxel mỗi khối. Ngoài khối lượng khổng lồ về dữ liệu, ta còn luôn phải đối mặt với các sai số do quá trình hô hấp gây ra, tạo chuyển động co giãn trên các bộ phận cơ thể mà ta đang quan tâm. Vấn đề này đã được trình bày trong hàng loạt hội thảo khoa học, với sự đóng góp của nhiều nhà khoa học: Sarut (2006) đưa ra công cụ phân tích ảnh chiếu xạ, Kessler (2006) có cách thức thu thập dữ liệu ảnh từ nhiều phương thức khác nhau, Guerrero, GZhang, Segar, Huang, Bilton (2007) với kỹ thuật đánh giá ảnh 4D [8],... Rất nhiều giải pháp kỹ thuật đưa ra đã dựa trên phương pháp của Horn và Schunck (đề xuất cho ảnh 2D từ 1981) để tính toán tốc độ của điểm ảnh trong luồng ảnh (chuyển động) [9]. Một số tác giả bổ sung thêm các ràng buộc và giả thiết để giải bài toán tối ưu [10]. Ngoài ra phương pháp của Lucass Kanade để khử nhiễu luồng ảnh 3D bằng phép tích chập với hàm phân bố Gauss cũng được phối hợp sử dụng [11]. Nhưng hầu hết các tác giả chỉ dừng lại ở xử lý ảnh 2D chịu nén, hoặc 3D không chịu nén và giải quyết bằng phương pháp lặp cho thuật toán Gauss-Seidel. Trong khuôn khổ bài báo, sẽ tiếp tục phát triển và phân tích kỹ hơn về mô hình bài toán xử lý ảnh 3D chịu nén, đồng thời đề xuất cấu trúc mạng nơron tế bào (CNN) 3 lớp 3D có thể tính được tốc độ luồng ảnh 3D với tốc độ cao. Các kết quả mới chỉ mang tính lý thuyết và sẽ tiếp tục được phát triển trong thời gian tiếp theo, hy vọng mang lại một phương pháp mới cho lĩnh vực xử lý ảnh động 4D-CT trong ngành y tế.

2. BÀI TOÁN TÍNH TỐC ĐỘ CỦA ẢNH CHỊU NÉN 4D

Khi phân tích ảnh động, một vấn đề quan trọng là phải xác định được tính tương thích

của các điểm ảnh giữa hai ảnh kế tiếp. Tốc độ của mỗi điểm ảnh trong chuỗi ảnh chuyển động (*Optical flow*) là một đặc trưng quan trọng để giải quyết vấn đề này. Đối với ảnh chụp cắt lớp cường độ ánh sáng của mỗi điểm ảnh $I(x, y, z, t)$ tỷ lệ với mật độ của các mô tế bào $\rho(x, y, z, t)$ trong đó x, y, z là tọa độ trong khối không gian Ω ta khảo sát và t là thời gian. Như vậy ta có thể mô tả về mặt toán học điểm ảnh 4D như sau

$$I(\mathbf{x}, t) \sim \rho(\mathbf{x}, t), \quad (1)$$

trong đó, $t \in [0, 1]$, $\mathbf{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \Omega$ và $\Omega \in \mathbb{R}^3$. Trong hầu hết các phương pháp xử lý ảnh cắt lớp người ta đều giả định rằng mật độ tế bào tương đương với cường độ ánh sáng của mỗi voxel không thay đổi theo thời gian $I(\mathbf{x}, t) \sim \rho(\mathbf{x}, t) = const$. Có nghĩa là

$$\frac{dI(\mathbf{x}, t)}{dt} = I_t + \nabla I^T \mathbf{v} = 0, \quad (2)$$

trong đó, $I_t = \frac{\partial I}{\partial t}$, và :

$$\nabla I = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \\ \frac{\partial I}{\partial z} \end{bmatrix} = \begin{bmatrix} I_x \\ I_y \\ I_z \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial t} \\ \frac{\partial z}{\partial t} \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}.$$

Phương trình (2) được gọi là phương trình bảo toàn khối lượng (conservation of mass) thể hiện tính liên tục hay gọi tắt là phương trình liên tục (continuity equation) của ảnh có cường độ không đổi theo thời gian hay còn gọi là ảnh không chịu nén (incompressible image). Trong trường hợp ảnh CT(Computed Tomography) lồng ngực, mật độ tế bào thay đổi từ ảnh này sang ảnh kia do có sự co nở của phổi trong quá trình hô hấp. Tuy nhiên, cho dù mật độ tế bào cục bộ không đồng đều ở phổi ta vẫn giả thiết được rằng tổng khối lượng của phổi không thay đổi trong quá trình hô hấp

$$\int_{\Omega} \rho(\mathbf{x}, 0) d\mathbf{x} = \int_{\Omega} \rho(\mathbf{x}, 1) d\mathbf{x}. \quad (3)$$

Lúc này phương trình bảo toàn khối lượng có dạng

$$I_t + \nabla I^T \mathbf{v} + I \operatorname{div}(\mathbf{v}) = 0, \quad (4)$$

trong đó,

$$\operatorname{div}(\mathbf{v}) = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}$$

là divergence của vectơ tốc độ điểm ảnh \mathbf{v} (*Optical Flow*). Phương trình (4) còn gọi là phương trình tốc độ của ảnh chịu nén (Compressible Image).

Để tính được tốc độ luồng ảnh $\mathbf{v} = \begin{bmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{bmatrix}$ với 3 ẩn số thì chỉ riêng phương trình tốc độ (2) hoặc (4) đều không đủ điều kiện. Do vậy ta cần tìm thêm các điều kiện ràng buộc

khác để có thể tính được vectơ tốc độ luồng ảnh \mathbf{v} một cách nhất quán. Áp dụng ý tưởng của Horn - Schunck [9], phối hợp phương trình (4) với giả thiết tốc độ của các voxel lân cận ít thay đổi vào trong một phiếm hàm (5) và tìm giá trị tối thiểu hóa phiếm hàm này

$$\min_{\mathbf{v}} \frac{1}{2} \int_{\Omega} \left\{ [I_t + \nabla I^T \mathbf{v} + I \operatorname{div}(\mathbf{v})]^2 + \alpha^2 \sum_{i=1}^3 \|\nabla v_i\|^2 \right\} d\mathbf{x}, \quad (5)$$

trong đó $\alpha > 0$ là hệ số làm trơn tự chọn. Ta có thể tìm được \mathbf{v} để tối thiểu hóa phiếm hàm (5) bằng phương pháp biến phân (*Variational Calculus*). Do (5) là một phiếm hàm lồi bình phương nên điều kiện cần và đủ của \mathbf{v} tối ưu là thoả mãn hệ phương trình Euler-Lagrange tương ứng

$$\frac{\partial f}{\partial \mathbf{x}} - \frac{d}{dt} \frac{\partial f}{\partial \mathbf{v}} = \mathbf{0}, \quad (6)$$

và điều kiện biên là:

$$(\nabla v_i + I h \mathbf{e}_i)^T \mathbf{n} = 0; i = 1, 2, 3. \quad (7)$$

Trong (6) và (7) ta sử dụng các ký hiệu

$$\begin{aligned} f &= h^2 + \alpha^2 \sum_{i=1}^3 \|\nabla v_i\|^2, \\ h &= I_t + \nabla I^T \mathbf{v} + I \operatorname{div}(\mathbf{v}), \end{aligned} \quad (8)$$

với \mathbf{e}_i là vectơ đơn vị của trục i , \mathbf{n} là vectơ pháp tuyến của mặt biên. Sau khi tính các đạo hàm của (6) và rút gọn, phương trình Euler-Lagrange có dạng

$$-I \nabla h - \alpha^2 \begin{bmatrix} \nabla^2 u \\ \nabla^2 v \\ \nabla^2 w \end{bmatrix} = \mathbf{0}. \quad (9)$$

Tiếp tục triển khai phương trình Euler Lagrange (9) ta được hệ phương trình (10) như sau

$$\begin{cases} I(I_{tx} + I_{xx}u + 2I_xu_x + I_yv + Iyv_x + I_zw + I_zw_x + Iu_{xx} + I_xv_y + Iv_{yx} + I_xw_z + Iw_{zx}) + \\ \alpha^2 \nabla_u^2 = 0 \\ I(I_{ty} + I_{xy}u + 2I_yu_y + I_{yy}v + I_xu_y + I_{zy}w + I_zw_y + I_yu_x + Iu_{xy} + Iv_{yy} + I_yw_z + Iw_{zy}) + \\ \alpha^2 \nabla_v^2 = 0 \\ I(I_{tz} + I_{xz}u + 2I_zw_z + I_{yz}v + I_xu_z + I_{zz}w + I_yv_z + I_zu_x + Iu_{xz} + Iv_{yx} + I_zv_y + Iw_{zz}) + \\ \alpha^2 \nabla_w^2 = 0 \end{cases} \quad (10)$$

trong đó,

$$\begin{aligned} \nabla^2 &\text{ là toán tử Laplace} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right); I_{tx} = \frac{\partial^2 I}{\partial t \partial x}; I_{xx} = \frac{\partial^2 I}{\partial x^2}; I_{yx} = \frac{\partial^2 I}{\partial y \partial x}; \\ U_{xx} &= \frac{\partial^2 U}{\partial x^2}; I_{ty} = \frac{\partial^2 I}{\partial t \partial y}; I_{xy} = \frac{\partial^2 I}{\partial x \partial y}; I_{yy} = \frac{\partial^2 I}{\partial y^2}; U_{yy} = \frac{\partial^2 U}{\partial y^2}; U_{xy} = \frac{\partial^2 U}{\partial x \partial y}; \end{aligned}$$

Để có thể tính được luồng ảnh \mathbf{v} ta cần giải hệ phương trình (10). Phương pháp số là phương pháp duy nhất hiện nay được dùng để giải bài toán optical flow. Thông thường người

ta rời rạc hóa các phép tính vi phân với các bước không gian chuẩn hóa $h_x = h_y = h_z = 1$ như sau

$$I_x = (I_{i+1,j,k} - I_{i-1,j,k})/2 ; I_y = (I_{i,j+1,k} - I_{i,j-1,k})/2 ; I_z = (I_{i,j,k+1} - I_{i,j,k-1})/2.$$

Toán tử Laplace ∇^2 được rời rạc hóa : $\nabla_u^2 = 6u_{i,j,k} - \overline{u_{i,j,k}}$, với

$$\overline{u_{i,j,k}} = (u_{i+1,j,k} + u_{i-1,j,k} + u_{i,j+1,k} + u_{i,j-1,k} + u_{i,j,k+1} + u_{i,j,k-1}).$$

∇_v^2 ; ∇_w^2 cũng được tính tương tự.

$$\begin{aligned} \frac{\partial u_{i,j}}{\partial x} &= \frac{1}{2}(u_{i+1,j} - u_{i-1,j}) = \Delta_x u_{i,j}; \quad \frac{\partial^2 u_{i,j}}{\partial x^2} = (2u_{i,j} - u_{i+1,j} - u_{i-1,j}) = \Delta_{xx} u_{i,j}, \\ \overline{u_{i,j}^*} &= \frac{1}{2}(u_{i+1,j} + u_{i-1,j}); \quad \overline{u_{i,j}} = \frac{1}{4}(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}), \\ \frac{\partial^2 u_{i,j}}{\partial x \partial y} &= \frac{1}{4}(u_{i+1,j+1} - u_{i+1,j-1} - u_{i-1,j+1} + u_{i-1,j-1}) = \Delta_{xy} u_{i,j}. \end{aligned}$$

Thay thế các phép tính xấp xỉ trên vào phương trình (10), ta có phương trình Euler-Lagrange dưới dạng rời rạc (11) như sau

$$\left\{ \begin{array}{l} I(I_{tx} + I_{xx}u_{i,j,k} + 2I_x\Delta_x u_{i,j,k} + I_{yx}v_{i,j,k} + I_y\Delta_x v_{i,j,k} + I_{zx}w_{i,j,k} + I_z\Delta_x w_{i,j,k} + \\ 2I(u_{i,j,k} - \overline{u_{i,j,k}^*}) + I_x\Delta_y v_{i,j,k} + I\Delta_{yx}v_{i,j,k} + I_x\Delta_z w_{i,j,k} + I\Delta_{zx}w_{i,j,k}) + 6\alpha^2(u_{i,j,k} - \overline{u_{i,j,k}}) = 0 \\ I(I_{ty} + I_{xy}u_{i,j,k} + 2I_y\Delta_y v_{i,j,k} + I_{yy}v_{i,j,k} + I_x\Delta_y u_{i,j,k} + I_{zy}w_{i,j,k} + I_z\Delta_y w_{i,j,k} + 2I(v_{i,j,k} - \overline{v_{i,j,k}^*}) \\ + I_y\Delta_x u_{i,j,k} + I\Delta_{xy}u_{i,j,k} + I_y\Delta_z w_{i,j,k} + I\Delta_{zy}w_{i,j,k}) + 6\alpha^2(v_{i,j,k} - \overline{v_{i,j,k}}) = 0 \\ I(I_{tz} + I_{xz}u_{i,j,k} + 2I_z\Delta_z w_{i,j,k} + I_{yz}v_{i,j,k} + I_x\Delta_z u_{i,j,k} + I_y\Delta_z v_{i,j,k} + I_{zz}w_{i,j,k} + 2I(w_{i,j,k} - \overline{w_{i,j,k}^*}) \\ + I_z\Delta_x u_{i,j,k} + I_z\Delta_y v_{i,j,k} + I\Delta_{xz}u_{i,j,k} + I\Delta_{yz}v_{i,j,k}) + 6\alpha^2(w_{i,j,k} - \overline{w_{i,j,k}}) = 0. \end{array} \right. \quad (11)$$

Bài toán tính tốc độ các điểm ảnh 4D-CT trở thành bài toán giải hệ phương trình Euler-Lagrange rời rạc để tìm các giá trị tốc độ của các điểm ảnh rời rạc $u_{i,j,k}$, $v_{i,j,k}$, $w_{i,j,k}$ trong khối không gian khảo sát Ω . Đây là một bài toán có độ tính toán phức tạp và rất lớn. Với ảnh 4D-CT có thể chứa tới 20 khối ảnh 3D cho với kích thước đạt tới $512 \times 512 \times 200$ voxel mỗi khối, ta có tổng số lượng voxel cần tính toán lên đến trên 100 triệu điểm ảnh. Đây là khối lượng dữ liệu khổng lồ mà các phương pháp xử lý ảnh thông thường trên các máy PC tuần tự khó có thể cho lời giải nhanh chóng, nhất là với các yêu cầu xử lý ảnh chuyển động thời gian thực. Bài báo đã đề xuất phương pháp xử lý song song sử dụng mạng nơon tế bào CNN để tăng tốc độ trong việc giải bài toán optical flow cho các ảnh 4D-CT chịu nén này.

3. PHƯƠNG PHÁP XÁC ĐỊNH TỐC ĐỘ LUỒNG ẢNH CHỊU NÉN 4D SỬ DỤNG MẠNG NƠN TẾ BÀO

Mạng nơon tế bào CNN (Cellular Neural Network) là hệ xử lý song song, đáp ứng yêu cầu về một hệ xử lý tốc độ cao, nhất là trong xử lý ảnh [4, 13]. Khối mạch cơ bản của CNN là tế bào (cell). Nó chứa các phần tử mạch tuyến tính và phi tuyến. Mỗi cell chỉ kết nối với các cell láng giềng. Các cell liền kề có thể ảnh hưởng trực tiếp đến nhau. Các cell không liên kết trực tiếp có thể tác động đến các cell khác gián tiếp bởi sự tác động lan truyền của mạng CNN. Mỗi cell $C_{i,j}$ có các tế bào láng giềng $C_{k,l}$. Các phương trình cơ bản mô tả hoạt động của mỗi tế bào của CNN một lớp 2D như sau:

+ Phương trình trạng thái:

$$\dot{x}_{ij} = -x_{ij} + z_{ij} + \sum_{C_{kl} \in S_{ij}(r)} A(i, j; k, l) \cdot y_{kl} + \sum_{C_{kl} \in S_{ij}(r)} B(i, j; k, l) \cdot u_{kl}. \quad (12)$$

+ Phương trình đầu ra:

$$y_{i,j} = f(x_{ij}) = \frac{1}{2} |x_{ij} + 1| + \frac{1}{2} |x_{ij} - 1|. \quad (13)$$

Ta thấy hệ phương trình cơ bản (12), (13) của mạng nơron tế bào là một hệ liên tục về thời gian và rời rạc về không gian. Do cấu trúc của các tế bào là như nhau nên rất thuận lợi cho việc thực hiện mạng trên chip. Chương trình xử lý trong CNN được thực hiện thông qua các lệnh mô tả sự kết nối cục bộ giữa các tế bào trong mạng **A**, **B**, **Z** (template). Mỗi lệnh ứng với một template. Trong thư viện của CNN có các template cơ bản và ngày càng được bổ sung phong phú hơn. Người lập trình còn có thể tự tìm cho mỗi bài toán những template đặc trưng tùy theo yêu cầu xử lý. Khi xử lý bài toán, chúng ta phải xây dựng mô hình toán học cho bài toán và tìm các template phù hợp. Để xác định tốc độ của mỗi voxel ta cần phải giải hệ phương trình Euler-Lagrange rời rạc (11). Ta có thể nhận được nghiệm của (11) bằng việc tìm nghiệm ổn định của hệ phương trình vi phân theo thời gian ảo τ sau

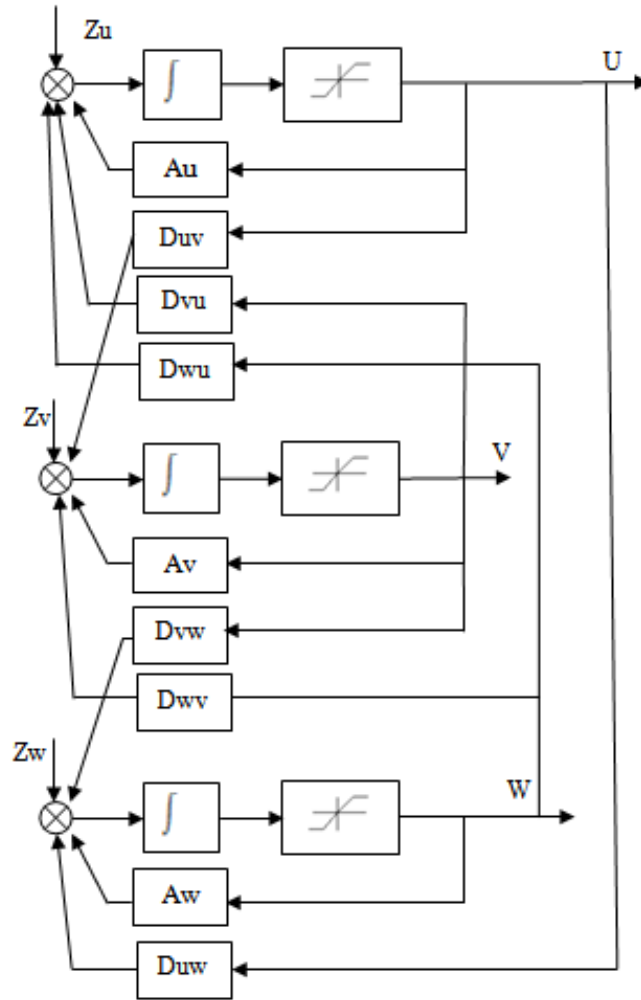
$$\begin{cases} -\frac{du_{i,j,k}}{d\tau} = I(I_{tx} + I_{xx}u_{i,j,k} + 2I_x\Delta_x u_{i,j,k} + I_{yx}v_{i,j,k} + I_y\Delta_x v_{i,j,k} + I_{zx}w_{i,j,k} + I_z\Delta_x w_{i,j,k} \\ + 2I(u_{i,j,k} - \overline{u_{i,j,k}^*}) + I_x\Delta_y v_{i,j,k} + I\Delta_{yx} v_{i,j,k} + I_x\Delta_z w_{i,j,k} + I\Delta_{zx} w_{i,j,k}) + 6\alpha^2(u_{i,j,k} - \overline{u_{i,j,k}}) \\ -\frac{dv_{i,j,k}}{d\tau} = I(I_{ty} + I_{xy}u_{i,j,k} + 2I_y\Delta_y v_{i,j,k} + I_{yy}v_{i,j,k} + I_x\Delta_y u_{i,j,k} + I_{zy}w_{i,j,k} + I_z\Delta_y w_{i,j,k} \\ + 2I(v_{i,j,k} - \overline{v_{i,j,k}^*}) + I_y\Delta_x u_{i,j,k} + I\Delta_{xy} u_{i,j,k} + I_y\Delta_z w_{i,j,k} + I\Delta_{zy} w_{i,j,k}) + 6\alpha^2(v_{i,j,k} - \overline{v_{i,j,k}}) \\ -\frac{dw_{i,j,k}}{d\tau} = I(I_{tz} + I_{xz}u_{i,j,k} + 2I_z\Delta_z w_{i,j,k} + I_{yz}v_{i,j,k} + I_x\Delta_z u_{i,j,k} + I_y\Delta_z v_{i,j,k} + I_{zz}w_{i,j,k} \\ + 2I(w_{i,j,k} - \overline{w_{i,j,k}^*}) + I_z\Delta_x u_{i,j,k} + I_z\Delta_y v_{i,j,k} + I\Delta_{xz} u_{i,j,k} + I\Delta_{yz} v_{i,j,k}) + 6\alpha^2(w_{i,j,k} - \overline{w_{i,j,k}}) \end{cases} \quad (14)$$

Ta xây dựng cấu trúc CNN giải hệ phương trình vi phân (14) trên như Hình . Đây là một mạng CNN 3 lớp 3D.

Lớp trên cùng để tính thành phần tốc độ $u_{i,j,k}$ của voxel (i, j, k) trong không gian 3D, lớp giữa tính thành phần tốc độ $v_{i,j,k}$ và lớp dưới để tính thành phần tốc độ $w_{i,j,k}$. Trạng thái của mạng là tốc độ voxel $\mathbf{v}(x, y, z)$. Hai ảnh 4D: $I(x, y, z, t), I(x, y, z, t + \Delta t)$ được dùng để tính các thành phần của các template phi tuyến của mạng. Giá trị ban đầu của các tích phân trong mạng là 0. Mạng được tính với các điều kiện cell biên là 0. Đầu ra của mạng là các giá trị của các thành phần tốc độ của voxel $\mathbf{v}(x(t), y(t), z(t))$. Ta thấy mối quan hệ giữa các lớp có tính phi tuyến với các giá trị cường độ ánh sáng. Trên cơ sở phân tích các mối liên kết giữa các thành phần của các voxel ở hệ phương trình (14) ta có thể xác định được các ma trận trọng số liên kết giữa các tế bào như sau $Z_u = I.I_{tx}$.

$$Au = \begin{bmatrix} \overline{A_{uz-1}} \\ \overline{A_{uz}} \\ \overline{A_{uz+1}} \end{bmatrix} \text{ với } \overline{A_{uz-1}} = \overline{A_{uz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\alpha^2 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{A_{uz}} = \begin{bmatrix} 0 & -I.I_x - I^2 - \alpha^2 & 0 \\ -\alpha^2 & I.I_{xx} + 6\alpha^2 + 2I^2 & -\alpha^2 \\ 0 & I.I_x - I^2 - \alpha^2 & 0 \end{bmatrix},$$



Hình 1. Mô hình CNN 3 lớp-3D tính tốc độ voxel

$$Dvu = \begin{bmatrix} \overline{D_{vuz-1}} \\ \overline{D_{vuz}} \\ \overline{D_{vuz+1}} \end{bmatrix}, \text{ với } \overline{D_{vuz-1}} = \overline{D_{vuz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{D_{vuz}} = \begin{bmatrix} \frac{I^2}{4} & \frac{-I \cdot I_y}{2} & \frac{-I^2}{4} \\ \frac{-I \cdot I_x}{2} & I \cdot I_{yx} & \frac{I \cdot I_x}{2} \\ \frac{-I^2}{4} & \frac{I \cdot I_y}{2} & \frac{I^2}{4} \end{bmatrix}, \quad Dwu = \begin{bmatrix} \overline{D_{wuz-1}} \\ \overline{D_{wuz}} \\ \overline{D_{wuz+1}} \end{bmatrix},$$

$$\overline{D_{wuz-1}} = \begin{bmatrix} 0 & \frac{I^2}{4} & 0 \\ 0 & \frac{-I \cdot I_x}{2} & 0 \\ 0 & \frac{-I^2}{4} & 0 \end{bmatrix},$$

$$\overline{D_{wuz+1}} = \begin{bmatrix} 0 & \frac{-I^2}{4} & 0 \\ 0 & \frac{I.I_x}{2} & 0 \\ 0 & \frac{I^2}{4} & 0 \end{bmatrix}, \quad \overline{D_{wuz}} = \begin{bmatrix} 0 & \frac{-I.I_z}{2} & 0 \\ 0 & I.I_{zx} & 0 \\ 0 & \frac{I.I_z}{2} & 0 \end{bmatrix},$$

$$Zv = I.I_{ty},$$

$$Av = \begin{bmatrix} \overline{A_{vz-1}} \\ \overline{A_{vz}} \\ \overline{A_{vz+1}} \end{bmatrix}, \quad \text{với } \overline{A_{vz-1}} = \overline{A_{vz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\alpha^2 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{A_{vz}} = \begin{bmatrix} 0 & -\alpha^2 & 0 \\ -I.I_y - I^2 - \alpha^2 & I.I_{yy} + 6\alpha^2 + 2I^2 & I.I_y - I^2 - \alpha^2 \\ 0 & -\alpha^2 & 0 \end{bmatrix},$$

$$Duv = \begin{bmatrix} \overline{D_{uwz-1}} \\ \overline{D_{uwz}} \\ \overline{D_{uwz+1}} \end{bmatrix}, \quad \text{với } \overline{D_{uwz-1}} = \overline{D_{uwz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{D_{uwz}} = \begin{bmatrix} \frac{I^2}{4} & \frac{-I.I_y}{2} & \frac{-I^2}{4} \\ \frac{-I.I_x}{2} & I.I_{xy} & \frac{I.I_x}{2} \\ \frac{-I^2}{4} & \frac{I.I_y}{2} & \frac{I^2}{4} \end{bmatrix}, \quad Duv = \begin{bmatrix} \overline{D_{vwz-1}} \\ \overline{D_{vwz}} \\ \overline{D_{vwz+1}} \end{bmatrix}, \quad \overline{D_{vwz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{-I^2}{4} & \frac{I.I_y}{2} & \frac{I^2}{4} \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{D_{vwz}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{-I.I_z}{2} & I.I_{zy} & \frac{I.I_z}{2} \\ 0 & 0 & 0 \end{bmatrix}, \quad \overline{D_{vwz-1}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{I^2}{4} & \frac{-I.I_y}{2} & \frac{-I^2}{4} \\ 0 & 0 & 0 \end{bmatrix}$$

$$Zw = I.I_{tz}$$

$$Aw = \begin{bmatrix} \overline{A_{wz-1}} \\ \overline{A_{wz}} \\ \overline{A_{wz+1}} \end{bmatrix} \overline{A_{wz-1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -I.I_z - I^2 - \alpha^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \overline{A_{wz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I.I_z - I^2 - \alpha^2 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{A_{wz}} = \begin{bmatrix} 0 & -\alpha^2 & 0 \\ -\alpha^2 & I.I_{zz} + 6\alpha^2 + 2I^2 & -\alpha^2 \\ 0 & -\alpha^2 & 0 \end{bmatrix}, \quad Duv = \begin{bmatrix} \overline{D_{vwz-1}} \\ \overline{D_{vwz}} \\ \overline{D_{vwz+1}} \end{bmatrix},$$

$$\overline{D_{vwz-1}} = \begin{bmatrix} 0 & \frac{I^2}{4} & 0 \\ 0 & \frac{-I.I_x}{2} & 0 \\ 0 & \frac{-I^2}{4} & 0 \end{bmatrix}, \quad \overline{D_{vwz+1}} = \begin{bmatrix} 0 & \frac{-I^2}{4} & 0 \\ 0 & \frac{I.I_x}{2} & 0 \\ 0 & \frac{I^2}{4} & 0 \end{bmatrix}, \quad \overline{D_{vwz}} = \begin{bmatrix} 0 & \frac{-I.I_z}{2} & 0 \\ 0 & I.I_{xz} & 0 \\ 0 & \frac{I.I_z}{2} & 0 \end{bmatrix},$$

$$Dvw = \begin{bmatrix} \overline{D_{vwz-1}} \\ \overline{D_{vwz}} \\ \overline{D_{vwz+1}} \end{bmatrix}, \quad \overline{D_{vwz-1}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{I^2}{4} & \frac{-I.I_y}{2} & \frac{-I^2}{4} \\ 0 & 0 & 0 \end{bmatrix},$$

$$\overline{D_{vwz+1}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{-I^2}{4} & \frac{I.I_y}{2} & \frac{I^2}{4} \\ 0 & 0 & 0 \end{bmatrix}, \quad \overline{D_{vwz}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{-I.I_z}{2} & I.I_{yz} & \frac{I.I_z}{2} \\ 0 & 0 & 0 \end{bmatrix}.$$

Mạng CNN 3 lớp 3D (Hình 1) với các templates phi tuyến dẫn dắt ở trên có thể giải được bài toán tính tốc độ luồng ảnh (Optical Flow) cho ảnh chịu nén 4D như ảnh 4D-CT lồng ngược. So với các phương pháp tính trước đây phương pháp này cho nhiều tính năng vượt trội. Thực tế, tốc độ tính toán của mạng CNN không phụ thuộc vào kích thước ảnh và chỉ phụ thuộc vào quá trình quá độ của mạch điện nên rất nhanh. Với công nghệ CMOS hiện nay tốc độ này khoảng $5-7\mu s$. Tốc độ này nhanh hơn nhiều so với các phương pháp tính trên máy PC kể cả các phương pháp lọc không gian-thời gian (Spatio-Temporal Filter). Nhược điểm của phương pháp đề xuất là mạng CNN có các trọng số kết nối phi tuyến đòi hỏi khi thiết kế mạch trên CMOS hoặc FPGA cần phải thiết kế rất thận trọng để bảo đảm mạng gọn, chạy ổn định và có độ chính xác cao.

4. KẾT LUẬN

Xử lý ảnh động cho các ảnh 4D-CT lồng ngược là một vấn đề thời sự do tính co giãn của các tế bào phổi trong quá trình hô hấp. Các giả thiết cường độ ánh sáng của các điểm ảnh CT không thay đổi theo thời gian như các phương pháp xử lý hiện nay đều gây nhiều sai số ở các ảnh chịu nén. Ở đây, ta đã khảo sát và xây dựng mô hình tốc độ luồng ảnh cho các ảnh 4D chịu nén trong ảnh CT lồng ngược nói riêng và các ảnh y tế nói chung. Các vấn đề liên quan như phương trình tốc độ luồng ảnh, tối thiểu hóa phiếm hàm theo mô hình Horn-Schunck, phương trình Euler-Lagrange rời rạc được áp dụng và phát triển cho ảnh chịu nén 4D-CT. Để tăng tốc độ tính toán bài báo đề xuất một mạng nơon tế bào 3 lớp 3D giải hệ phương trình Euler-Lagrange rời rạc của ảnh 4D-CT để tính được tốc độ của các voxel từ 2 ảnh kế tiếp. Đây là một phương pháp mới xử lý song song cho tốc độ tính toán rất cao xuất phát từ đặc tính của mạng nơon tế bào và khả năng cứng hóa mạng trên nền chip FPGA. Có thể tích hợp thêm với một số phương pháp và kỹ thuật khác nhằm giảm nhiễu, cải thiện chất lượng cho các ảnh thu được từ scan 4D-CT. Vấn đề này sẽ được đề cập đến trong một bài báo khác, với hy vọng có thể đi đến hoàn thiện hơn kỹ thuật xử lý ảnh động chịu nén trong y tế khi sử dụng công nghệ CNN.

TÀI LIỆU THAM KHẢO

- [1] J.Barron and N.Thacker, Tutorial:Computing 2D and 3D optical flow, Tina Memo No.2004-012, *Internal document of Imaging Science and Biomedical Engineering Division*, Medical School, University of Manchester, 2004.
- [2] R. Adrian, Twenty year of particle image velocimetry, *Experiments in Fluid* **39** (2) (2005) 159–169.
- [3] S.Beauchemin and J. Barron, The computation of optical flow, *ACM Computing Surveys (ACMCS1995)* **27**(3) (September,1995) 433–467.
- [4] Angela Slavova, *Cellular Neural Networks: Dynamics and Modelling*, Kluwer Academic Publishers, 2005.
- [5] F.Meyer, Iterative Image Transformation for an Automatic Screening of Cervical Cancer, *Journal of Histochemistry and Cytochemistry* **27**(1) (1979) 128–135.

- [6] Yan Ha, Method of edge detection based on Non-Linear Cellular Automata, *Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongqing, China, 25-27 June 2008 (6817–6821).
- [7] T.Corpetti, E.Memin, A.Santa-Cruz, D.Heitz, and G.Arroyo, Optical flow estimation in experimental fluid mechanics, *Proceedings of the 7th International Symposium on Signal Processing and Its Applications* **1**, Paris, France, July 2003 (633–636).
- [8] E.Castillo, Yin Zang, and T.Guerrero, Compressible image registration for thoracic computed tomography image, *Journal of Medical and Biological Engineering* **29**(5) (2009) 222–233.
- [9] B.Horn and Schunck, Determining optical flow, *Artificial Intelligence*, **17** (August 1981) 185–203.
- [10] R.Wildes, M.Amabile, Ann-Marie Lanzillotto, and Tzong-Shyng Leu, Recovering estimates of fluid flow from image sequence data, *Computer Vision and Image Understanding* **80** (2000) 246-266.
- [11] B.D. Lucas and T. Kanade, An iterative image registration technique with an application to stereo vision, *Proceedings of the DARPA Image Understanding Workshop*, Washington D.C., USA, April 1981, 121–130.
- [12] A.Bruhn and W.Joachim , Combining Local and Global Optical flow methods, *International Journal of computer Vision* **61**(3) (2005) 211–231.
- [13] Leon O. Chu and T. Roska, *Cellular neural networks and visual computing: foundation and applications*, Cambridge University Press, 2002, USA.

Nhận bài ngày 25 - 10 - 2010