

**PHƯƠNG PHÁP NÉN ẢNH SỬ DỤNG MẠNG NƠON NHÂN TẠO VÀ K-MEANS****IMAGE COMPRESSION BASED ON ARTIFICIAL NEURAL NETWORKS AND K-MEANS****Võ Văn Nhật, Phạm Minh Tuấn***Trường Đại học Bách khoa, Đại học Đà Nẵng; Email: v2nhat@gmail.com, pmtuan@dut.udn.vn*

**Tóm tắt** – Sử dụng mạng nơon nhân tạo là một phương pháp hiệu quả trong việc nén ảnh. Mạng nơon nhân tạo có khả năng xấp xỉ không gian màu của một bức ảnh bằng một không gian nhỏ hơn so với không gian của bức ảnh ban đầu. Nếu ảnh đầu vào có các dạng màu sắc gần giống nhau tại các vị trí khác nhau trên cùng một bức ảnh thì việc xấp xỉ sẽ dễ dàng. Tuy nhiên, ảnh đầu vào có rất nhiều dạng màu sắc khác nhau thì việc xấp xỉ sẽ trở nên khó khăn. Báo cáo này đề xuất phương pháp nén ảnh sử dụng mạng neural kết hợp với phương pháp phân nhóm k-means nhằm hạn chế sự mất mát thông tin màu sắc của bức ảnh trong quá trình nén. Trước tiên, phương pháp đề xuất chia bức ảnh thành nhiều block khác nhau. Sau đó phân nhóm các block này sử dụng k-means. Mỗi nhóm block sẽ được thông qua một mạng nơon khác nhau để xây dựng không gian xấp xỉ. Kết quả thực nghiệm trên các ảnh thực cho thấy phương pháp đề xuất tốt hơn so với phương pháp trước đó.

**Từ khóa** – mạng nơon nhân tạo; phân nhóm; k-mean; nén ảnh; ảnh số.

**Abstract** – Using artificial neural network is an effective method in image compression. Artificial neural networks have the ability to approximate the color space of an image by a smaller space than from the original image. The approximation will be easy if the input image has many similarities in color at different locations. However, input image has many different types of colors, the approximation becomes difficult. This paper proposes the image compression method using a neural network method combined with k-means to minimize the loss of color information of the image in the compression process. First, the proposed method split image into different blocks. Then cluster these blocks using k-means. Finally, this paper builds an approximation space using the neural networks for all groups of blocks. Experimental results on real images show that the proposed method is better than the conventional method.

**Key words** – neural networks; clustering; k-mean; compress; image.

**1. Đặt vấn đề**

Với sự bùng nổ của công nghệ thông tin, nhu cầu lưu trữ dữ liệu luôn là mối quan tâm hàng đầu của các nhà nghiên cứu hiện nay. Cùng với các giải pháp phần cứng như chế tạo các thiết bị lưu trữ có kích thước nhỏ đồng thời có dung lượng lớn, thì các giải pháp phần mềm như nén dữ liệu cũng được nghiên cứu nhằm giải quyết bài toán lưu trữ. Ngoài lợi ích về lưu trữ thì giải pháp nén dữ liệu còn giải quyết được tốc độ truyền dẫn hay ghi đọc vì các dữ liệu đã được nén này có kích thước nhỏ hơn so với dữ liệu ban đầu.

Báo cáo này tập trung nghiên cứu phương pháp nén ảnh [1] sử dụng học máy. Đặc biệt là phương pháp nén ảnh sử dụng mạng nơon nhân tạo [2][3][4][5]. Phương pháp nén ảnh sử dụng mạng nơon nhân tạo có thể xấp xỉ không gian màu của một bức ảnh bằng một không gian nhỏ hơn so với không gian của bức ảnh ban đầu. Các không gian xấp xỉ này chính là các tín hiệu đầu vào của các nơon đầu ra trong mạng nơon nhân tạo. Nhờ vậy chỉ cần lưu trữ các trạng thái của các nơon lớp đầu ra này ta có thể tiết kiệm được chi phí lưu trữ mà không làm mất đi nhiều lượng thông tin màu sắc trong bức ảnh. Vấn đề được đặt ra là việc xấp xỉ thông qua mạng nơon nhân tạo sẽ chịu ảnh hưởng trực tiếp bởi thông tin màu sắc của ảnh đầu vào. Nếu ảnh đầu vào có các dạng màu sắc gần giống nhau, nghĩa là màu sắc tại các vị trí khác nhau trên cùng một bức ảnh có nhiều sự tương đồng thì việc xấp xỉ sẽ dễ dàng. Tuy nhiên, ảnh đầu vào có rất nhiều dạng màu sắc khác nhau thì việc xấp xỉ sẽ trở nên khó khăn. Dẫn tới sự mất mát thông tin của bức ảnh qua quá trình nén bằng mạng nơon.

Nghiên cứu của bài báo này đề xuất phương pháp nén ảnh sử dụng mạng nơon kết hợp với phương pháp phân nhóm k-means nhằm hạn chế sự mất mát thông tin màu sắc của bức ảnh trong quá trình nén. Trước tiên, phương pháp đề xuất chia bức ảnh thành nhiều block khác nhau. Sau đó

phân nhóm các block này sử dụng phương pháp k-means. Mỗi nhóm block đã được phân nhóm sẽ được thông qua một mạng nơon khác nhau để xây dựng không gian xấp xỉ. Dễ dàng thấy rằng các block có cùng một nhóm sẽ có màu sắc tương đồng nên việc xấp xỉ sẽ dễ dàng hơn so với việc không phân cụm, dẫn tới hiệu quả nén ảnh tốt hơn.

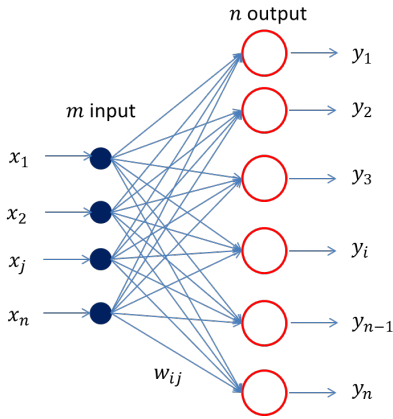
**2. Phương pháp nén ảnh sử dụng mạng nơon****2.1. Mạng nơon**

Mạng nơon nhân tạo là mô hình tính toán được xây dựng dựa trên mạng nơon sinh học. Một mạng nơon nhân tạo bao gồm nhiều tế bào nơon kết nối với nhau theo một cách nào đó. Một số tế bào trong mạng đảm nhận nhiệm vụ nhận thông tin từ bên ngoài được gọi là các nơon đầu vào. Một số tế bào nhận nhiệm vụ trả về giá trị tính toán của mạng gọi là các nơon đầu ra. Hầu hết các mạng nơon nhân tạo có hệ thống tự thích ứng (adaptive system), nghĩa là có chức năng tự thay đổi các thông số của mạng dựa trên các thông tin đầu vào và đầu ra, hay các dòng thông tin đi qua mạng trong quá trình học. Hiện nay, có rất nhiều cấu trúc mạng nơon được nghiên cứu nhằm giải quyết các bài toán cụ thể khác nhau. Nghiên cứu này chỉ đề cập đến mạng nơon truyền thẳng (feed-forward) với phương pháp học lan truyền ngược (back-propagation). Tín hiệu cấu trúc mạng này được truyền thẳng từ một hoặc một số nơon đầu vào qua một số nơon trung gian và cuối cùng là nơon đầu ra, trong mạng không xuất hiện chu trình trong đường đi của tín hiệu.

**2.1.1. Mạng nơon truyền thẳng một lớp**

Mô hình mạng nơon truyền thẳng một lớp là mô hình liên kết cơ bản và đơn giản nhất trong các loại mạng nơon nhân tạo. Tất cả các nơon đều mang cả hai chức năng nhận và xuất tín hiệu. Các nơon này được tổ chức thành một hàng và tín hiệu chỉ được truyền theo một hướng nhất định. Các

tín hiệu đầu vào được truyền vào các nơon đã được sắp xếp theo các trọng số khác nhau. Sau quá trình tính toán, các nơon này cho ra một danh sách các tín hiệu đầu ra thể hiện như Hình 1.



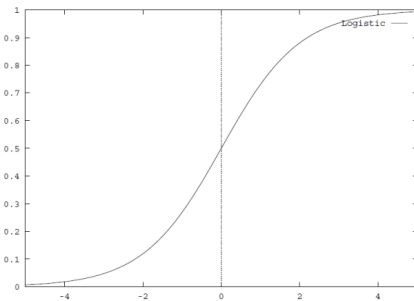
**Hình 1:** Cấu trúc mạng nơon truyền thẳng 1 lớp

Khi ta có một mạng nơon truyền thẳng một lớp với  $m$  tín hiệu đầu vào và  $n$  tín hiệu đầu ra thì ứng với mỗi vector đầu vào  $x = (x_1, \dots, x_m)^T$  ta có một vector đầu ra  $o = (o_1, \dots, o_n)^T$ . Các vector đầu ra của mỗi tế bào nơon được xác định theo công thức sau:

$$o_i = f \left( b_i + \sum_{j=1}^m w_{ij} x_j \right), \forall i \in \{1, \dots, n\}$$

Với  $w_{ij}$  là trọng số của tín hiệu đầu vào thứ  $j$  đối với tế bào nơon  $i$ .  $b_i$  là ngưỡng hay còn gọi là bias của tế bào nơon  $i$ .  $f(\cdot)$  là hàm kích hoạt trong nhân của tế bào nơon. Hàm kích hoạt là hàm logistic được xác định như sau:

$$f(\text{net}) = \frac{1}{1 + e^{-\text{net}}}$$



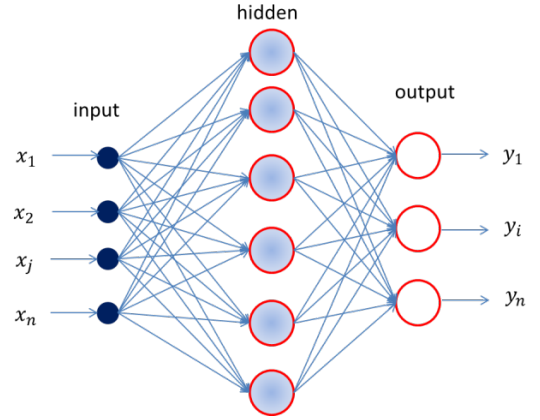
**Hình 2:** Hàm sigmoid

Hình 2 biểu diễn đồ thị hàm logistic. Ta dễ dàng nhận thấy hàm này sẽ có giá trị trong khoảng 0 đến 1. Vì vậy mỗi tế bào nơon thường dùng để phân loại nhị phân một cách tuyến tính cho một tập dữ liệu đầu vào nào đó.

### 2.1.2. Mạng nơon truyền thẳng nhiều lớp

Mạng nơon truyền thẳng một lớp thường chỉ dùng để phân loại hay xấp xỉ tuyến tính. Đối với những bài toán phi tuyến tính thì mạng nơon truyền thẳng một lớp còn nhiều hạn chế. Một trong những hướng giải quyết cho những hạn

chế của mạng nơon truyền thẳng một lớp đó là mạng nơon truyền thẳng nhiều lớp. Mạng này được tạo nên từ nhiều mạng nơon truyền thẳng một lớp bằng cách liên kết với nhau theo thứ tự liên tiếp nhất định. Bất kì 2 nơon thuộc 2 lớp liên tiếp đều được nối với nhau bằng một kết nối, và trong mỗi kết nối các tín hiệu đầu ra của lớp trước chính là tín hiệu đầu vào của lớp sau. Trong mạng nơon nhiều lớp, các nơon trong cùng một lớp không được kết nối với nhau. Mạng nơon nhiều lớp có thể hiện thực hiện việc phân loại hay xấp xỉ phi tuyến tính. Hình 3 thể hiện một mạng nơon truyền thẳng 2 lớp, gồm lớp nơon ẩn và lớp nơon đầu ra. Lưu ý rằng một số tài liệu gọi mạng nơon ở Hình 3 này là mạng nơon 3 lớp vì coi các tín hiệu đầu vào như là một lớp.



**Hình 3:** Một ví dụ về mạng nơon truyền thẳng 2 lớp

### 2.1.3. Phương pháp huấn luyện

Việc huấn luyện mạng nơon truyền thẳng 2 lớp đồng nghĩa với việc tìm các trọng số  $w$  tham số  $b$  của tất cả tế bào nơon. Ở đây, để dễ cho việc tính toán, tham số  $b$  của tế bào nơon được xem như một trọng số của  $w$  với tín hiệu đầu vào là 1. Khi đó, hàm tối ưu trong quá trình luyện tập được định nghĩa như sau:

$$E = \frac{1}{2} \sum_{j=1}^N (y_j - o_j)^2$$

Với  $y_j$  là tín hiệu huấn luyện (tín hiệu đầu ra thực tế) của lớp  $j$  ứng với dữ liệu đầu vào.  $o_{pj}$  là tín hiệu đầu ra của tế bào nơon thứ  $j$  đối với dữ liệu đầu vào. Theo thuật toán rơi dốc nhanh nhất (Steepest Descent Method [6]) thì lượng chỉnh sửa trong mỗi lần update trọng số  $w$  là:

$$\Delta w_{ji} = -\alpha \frac{\partial E}{\partial w_{ji}} = -\alpha \frac{\partial E}{\partial \text{net}_j} \frac{\partial \text{net}_j}{\partial w_{ji}}$$

$$\text{Trong đó, } \text{net}_j = \sum_k o_k w_{jk}$$

Với,  $o_k$  là tín hiệu đầu ra của tế bào nơon thứ  $j$ .  $w_{ji}$  là trọng số của tế bào nơon thứ  $j$  được kết nối từ tín hiệu đầu ra của tế bào nơon thứ  $i$ . Ta lại có:

$$\frac{\partial \text{net}_j}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \sum_k o_k w_{jk} = o_i$$

Và,

$$\frac{\partial E}{\partial \text{net}_j} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial \text{net}_j}$$

Với hàm hoạt động  $f$  là hàm logistic thì:

$$\frac{\partial o_j}{\partial \text{net}_j} = f'(\text{net}_j) = o_j (1 - o_j)$$

+ Trong trường hợp tế bào nơron thứ  $j$  thuộc lớp đầu ra thì:

$$\frac{\partial E}{\partial o_j} = -(y_j - o_j)$$

Suy ra:

$$\Delta w_{ji} = \alpha (y_j - o_j) o_j (1 - o_j) o_i$$

+ Trong trường hợp tế bào nơron thứ  $j$  thuộc lớp ẩn thì:

$$\begin{aligned} \frac{\partial E}{\partial o_j} &= \sum_k \frac{\partial E}{\partial \text{net}_k} \frac{\partial \text{net}_k}{\partial o_j} \\ &= \sum_k \frac{\partial E}{\partial \text{net}_k} \frac{\partial}{\partial o_j} \sum_i o_i w_{ki} \\ &= \sum_k \frac{\partial E}{\partial \text{net}_k} w_{kj} \end{aligned}$$

Suy ra :

$$\Delta w_{ji} = \alpha \left( o_j (1 - o_j) \sum_k \frac{\partial E}{\partial \text{net}_k} w_{kj} \right) o_i$$

Tóm lại, thuật toán tìm trọng số  $w$  được trình bày như sau:

**Bước 1:** Khởi tạo  $w \in \mathbb{R}^d$  một cách ngẫu nhiên cho tất cả các tế bào nơron.

**Bước 2:** Ứng với bộ dữ liệu huấn luyện, lặp đi lặp lại các bước sau cho tới khi  $w$  của tất cả các tế bào nơron hội tụ:

+ Tính tất cả các tín hiệu đầu ra của tất cả tế bào nơron trong mạng,

$$o_j = \frac{1}{1 + e^{-\left(\sum_k o_k w_{jk}\right)}}$$

+ Cập nhật giá trị của  $w$ ,

Đặt:

$$\delta_j = (y_j - o_j) o_j (1 - o_j)$$

- Trong trường hợp tế bào nơron thứ  $j$  thuộc lớp đầu ra thì trọng số của nhánh từ tế bào nơron thứ  $i$  là:

$$\Delta w_{ji} = \alpha \delta_j o_i$$

- Trong trường hợp tế bào nơron thứ  $j$  thuộc lớp ẩn thì trọng số của nhánh từ tế bào nơron thứ  $i$  là:

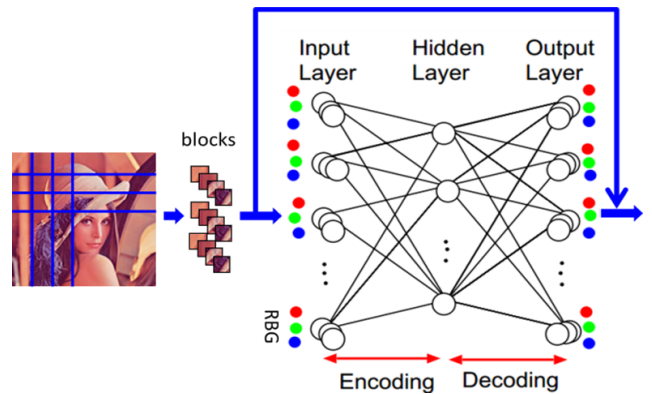
$$\Delta w_{ji} = \alpha \left( o_j (1 - o_j) \sum_k \delta_k w_{kj} \right) o_i$$

Trong đó  $k$  là số thứ tự của tế bào nơron nhận đầu ra của  $j$  làm tín hiệu đầu vào.

+ Cập nhật:  $w'_{ji} = w_{ji} + \Delta w_{ji}$

## 2.2. Phương pháp nén ảnh sử dụng mạng nơron

Trong mô hình mạng truyền thẳng 2 lớp, khi vector tín hiệu đầu vào và đầu ra bằng nhau, số lượng tế bào nơron lớp ẩn  $n_2$  nhỏ hơn tế bào nơron đầu ra  $n_1$  thì ta có thể thu được thông tin đã được nén của các tín hiệu đầu vào bằng cách lưu trữ các hiệu đầu ra của lớp ẩn. Phương pháp nén ảnh đề cập trong bài báo này chính là việc sử dụng mô hình mạng nơron 2 lớp trong bài toán tìm các trọng số của mạng nơron sao cho các tín hiệu ảnh đầu vào và đầu ra là như nhau. Qua quá trình huấn luyện, các trọng số của các nơron sẽ được cập nhật sao cho tín hiệu đầu ra của mạng xấp xỉ với tín hiệu ảnh đầu vào. Từ đó ta chỉ cần lưu lại các trọng số và các tín hiệu đầu vào của lớp nơron đầu ra là có thể tái hiện được thông tin của ảnh cần được nén.



Hình 4: Nén ảnh sử dụng mạng nơron 2 lớp

Hình 4 biểu diễn mô hình nén ảnh thông qua mạng nơron 2 lớp. Đầu tiên, mô hình này chia bức ảnh cần nén thành các block cùng một kích thước định trước  $B_x \times B_y$ . Các block này sẽ được sử dụng như một tập huấn luyện mà trong đó mỗi block vừa đóng vai trò làm tín hiệu đầu vào, vừa đóng vai trò làm tín hiệu đầu ra. Gọi là số tế bào của lớp ẩn và  $B = 3 \times B_x \times B_y$  là số tín hiệu đầu vào của một block thì ta có công thức tỉ lệ nén như sau:

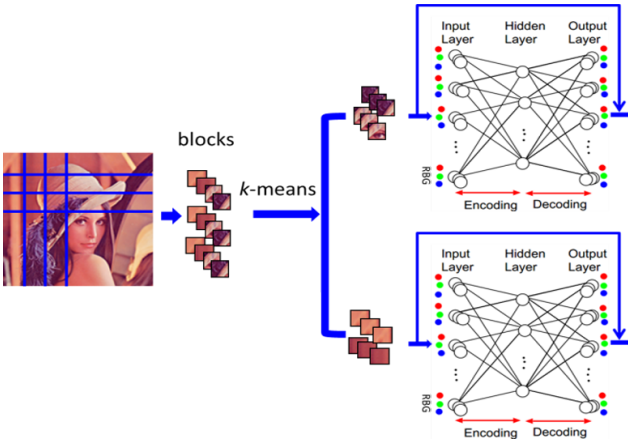
$$C = \frac{H}{B} = \frac{H}{3 \times B_x \times B_y}$$

## 3. Phương pháp đề xuất

### 3.1. Mô hình đề xuất

Trong mô hình nén ảnh đã đề cập ở trên, tất cả các block đều được sử dụng trên một mạng nơron. Vì vậy các block có màu sắc gần giống nhau thì việc cập nhật các trọng số của mạng sẽ dễ dàng và chính xác. Tuy nhiên, các block có màu sắc khác nhau thì việc huấn luyện sẽ trở nên khó khăn. Dẫn tới sự mất mát thông tin màu sắc của bức ảnh qua quá trình nén bằng mạng nơron. Bài báo này đề xuất phương pháp nén ảnh sử dụng mạng nơron kết hợp với phương pháp phân nhóm k-means nhằm giảm sự mất mát thông tin màu sắc của bức ảnh trong quá trình nén.

Hình 5 biểu diễn mô hình đề xuất. Đóng góp chính của mô hình này là đề xuất việc phân nhóm cho các blocks bằng phương pháp k-means. Sau đó mỗi nhóm sẽ là tập huấn luyện cho một mạng nơron riêng biệt.



**Hình 5:** Phương pháp đề xuất. Các blocks được phân nhóm bằng thuật toán k-means sau đó mỗi nhóm sẽ là tập huấn luyện cho từng mạng nơron

### 3.2. Phương pháp phân cụm k-means

K-means [7][8] là phương pháp phân lớp hoặc gán nhãn cho các đối tượng chưa được gán nhãn cho trước. K-means hay còn gọi là hard c-means (HCM) chính là thuật toán đi tìm lời giải cho bài toán tối ưu sau:

$$\min_{u,c} \sum_{i=1}^N \sum_{j=1}^C u_{ij} d(i,j)$$

$$\text{s.t.} \sum_{j=1}^C u_{ij} = 1, u_{ij} \in \{0,1\}$$

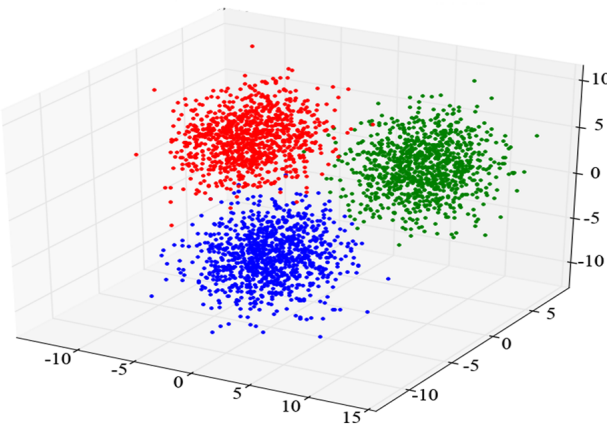
Trong đó,

$$d(i,j) = \|x_i - c_j\|^2$$

là cự ly bình phương giữa dữ liệu  $x_i$  và vector  $c_j$  đại diện cho lớp có nhãn  $j$ . Và  $u_{ij}$  là mức độ phụ thuộc của  $x_i$  trong lớp có nhãn  $j$ .

Đầu ra của các vector  $c_j$  trong thuật toán k-means và fuzzy c-means được tính bởi công thức sau:

$$c_j = \frac{\sum_{i=1}^N u_{ij} x_i}{\sum_{i=1}^N u_{ij}}$$



**Hình 6:** Ví dụ kết quả phân nhóm sử dụng k-means

Hình 6 là ví dụ thể hiện kết quả phân nhóm sử dụng phương pháp k-means. Mỗi màu tương ứng với mỗi nhóm dữ liệu đã được gán nhãn bởi k-means.

### 3.3. Các thuật toán trong mô hình đề xuất

Các bước trong thuật toán nén ảnh và giải nén được trình bày như sau:

#### 3.3.1. Thuật toán nén ảnh

**Bước 1:** Chia ảnh ra làm các block.

**Bước 2:** Phân các block thành k nhóm bằng phương pháp k-means.

**Bước 3:** Khởi tạo k mạng nơron.

**Bước 4:** Cập nhật trọng số  $w$  của từng mạng nơron ứng với các nhóm block.

**Bước 5:** Kiểm tra tỷ lệ lỗi của ảnh đầu ra so với ảnh gốc, nếu đạt yêu cầu thì đến Bước 6, nếu không quay về Bước 4.

**Bước 6:** Lưu tất giá trị đầu vào và trọng số của các nơron thuộc lớp đầu ra ứng với từng block. Gọi dữ liệu này là dữ liệu nén.

#### 3.3.2. Thuật toán giải nén

**Bước 1:** Đọc dữ liệu nén.

**Bước 2:** Khởi tạo mạng nơron 1 lớp.

**Bước 3:** Tái tạo giá trị màu của các block bằng cách tính giá trị đầu ra của các nơron ứng với giá trị đầu vào và trọng số được lưu trong file nén.

**Bước 4:** Kết nối các block để tạo thành ảnh ban đầu.

## 4. Kết quả thực nghiệm

Báo cáo này tiến hành so sánh phương pháp đề xuất và phương pháp mạng nơron trước đó để nén ảnh các ảnh ở Hình 7.



**Hình 7:** Ảnh gốc sử dụng trong việc nén ảnh

Báo cáo này sử dụng Mean Square Error (MSE) [9] để tính toán độ lệch màu trong quá trình nén ở cả hai phương pháp và lấy đó làm tiêu chí để so sánh. Độ lệch màu được tính theo công thức sau:

$$T_{\text{error}} = \frac{1}{I_w \times I_H} \sum_{x=1}^{I_w} \sum_{y=1}^{I_H} \|f(x,y) - \hat{f}(x,y)\|$$

Với  $I_w \times I_H$  là kích thước của ảnh ban đầu.  $f(x,y)$  và  $\hat{f}(x,y)$  lần lượt là vector của 3 màu đỏ, xanh lá cây, xanh da trời tại điểm ảnh  $(x,y)$  trước nén và sau khi nén. Trong bài báo này, chọn 8 loại kích thước block (3x3 đến 10x10) và 3 loại tỷ lệ nén (40%, 60% và 80%) để tiến hành thực nghiệm. Đối với phương pháp đề xuất, báo cáo cố định số lượng nhóm của k-means là 5. Bảng 1 là kết quả so sánh độ lệch màu trung bình trong quá trình nén giữa hai phương pháp. Ta dễ dàng nhận thấy rằng phương pháp đề xuất có độ lệch màu ít hơn hẳn so với phương pháp trước đó là chỉ dùng mạng nơron.

**Bảng 1:** Độ lệch màu trong quá trình nén

| Kích thước block | Tỷ lệ nén | Phương pháp trước đó | Phương pháp đề xuất |
|------------------|-----------|----------------------|---------------------|
| 3x3              | 40%       | 3.896                | 0.120               |
| 3x3              | 60%       | 3.753                | 0.133               |
| 3x3              | 80%       | 3.578                | 0.099               |
| 4x4              | 40%       | 4.558                | 0.188               |
| 4x4              | 60%       | 4.424                | 0.117               |
| 4x4              | 80%       | 4.296                | 0.114               |
| 5x5              | 40%       | 4.716                | 0.300               |
| 5x5              | 60%       | 4.746                | 0.255               |
| 5x5              | 80%       | 4.588                | 0.275               |
| 6x6              | 40%       | 5.013                | 0.523               |
| 6x6              | 60%       | 5.103                | 0.656               |
| 6x6              | 80%       | 5.229                | 0.667               |
| 7x7              | 40%       | 5.290                | 0.814               |
| 7x7              | 60%       | 5.213                | 1.150               |
| 7x7              | 80%       | 5.169                | 1.115               |
| 8x8              | 40%       | 5.497                | 1.196               |
| 8x8              | 60%       | 5.313                | 1.892               |
| 8x8              | 80%       | 5.745                | 1.782               |
| 9x9              | 40%       | 6.112                | 1.555               |
| 9x9              | 60%       | 6.399                | 1.892               |
| 9x9              | 80%       | 6.815                | 2.458               |
| 10x10            | 40%       | 6.462                | 1.706               |
| 10x10            | 60%       | 6.816                | 2.214               |
| 10x10            | 80%       | 7.378                | 3.087               |

**Hình 8:** Ảnh qua quá trình nén và giải nén sử dụng NN không có k-means**Hình 9:** Ảnh qua quá trình nén và giải nén sử dụng phương pháp đề xuất

Hình 8 và Hình 9 lần lượt là ảnh đã qua quá trình nén và giải nén sử dụng phương pháp trước đó và phương pháp đề xuất với kích thước block là 3x3 và tỷ lệ nén là 40%. Ta có thể nhận thấy rằng, với dùng một tỷ lệ nén như nhau phương pháp đề xuất cho ra ảnh gần giống với ảnh gốc hơn so với phương pháp trước đó.

## 5. Kết luận

Bài báo này đã trình bày phương pháp nén ảnh sử dụng mạng nơron nhân tạo. Sau đó đã trình bày sơ lược mạng nơron nhân tạo và cách sử dụng mạng này để nén ảnh. Tiếp theo báo cáo đề xuất phương pháp kết hợp với phương pháp phân nhóm k-means nhằm nâng cao chất lượng ảnh trong quá trình nén. Kết quả thực nghiệm cho thấy độ lệch màu trong quá trình nén và giải nén của phương pháp đề xuất ít hơn hẳn so với phương pháp trước đó. Điều đó cho thấy sự hiệu quả của phương pháp đề xuất khi sử dụng phương pháp phân cụm k-means trước khi sử dụng mạng nơron để nén ảnh.

## Tài liệu tham khảo

- [1] K. R. Rao and P. Yip, "Discrete Cosine Transform Algorithms", Advantages, Applications, Academic Press, 1990.
- [2] K. R. Rao, P. Madhusudana, S. Nachiketh and S.S.Keerthi", Image compression using artificial neural", *IEEE, ICMLC* 2010, pp. 121-124, 2010.
- [3] D. Dutta, S. Choudhury, M. Hussain and S. Majumder, "Digital image compression using neural network", *IEEE, International Conference on Advances in Computing, Control, Telecommunication Technologies*, p. ACT '09, 2009.
- [4] N.M.Rahim and T.Yahagi, "Image Compression by new sub-image bloc Classification techniques using Neural Networks", *IEICE Trans. On Fundamentals*, Vols. E83-A, no. 10, pp. 2040-2043, 2000.
- [5] M. S. Rahim, "Image compression by new sub- image block Classification techniques using neural network", *IEICE Trans. On Fundamentals of Electronics, Communications, and Computer Sciences*, Vols. E83-A, no. 10, pp. 2040- 2043, 2000.
- [6] K. C. Kiwiel, "Convergence and efficiency of subgradient methods for quasiconvex minimization", *Mathematical Programming (Berlin, Heidelberg: Springer)*, p. 1-25, 2001.
- [7] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations", *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, p. 281-297, 1967.
- [8] J. Bezdek, R. Ehrlich and W. Full, "FCM: the fuzzyc-means clustering algorithm", *Computers and Geosciences*, vol. 10, p. 191-203, 1984.
- [9] D. Wackerly and W. Scheaffer, "Mathematical Statistics with Applications", Belmont, CA, USA: Thomson Higher Education, vol. 7, 2008.

(BBT nhận bài: 22/12/2013, phản biện xong: 27/12/2013)