

MỘT PHƯƠNG PHÁP PHÂN HOẠCH DỮ LIỆU TRÊN CÂY PHÂN CẤP CHO BÀI TOÁN TÌM KIẾM ẢNH THEO NỘI DUNG

Nguyễn Thị Thùy Trang¹, Nguyễn Minh Hải², Nguyễn Văn Tùng¹, Nguyễn Văn Thịnh¹, Văn Thế Thành^{3*}

¹Khoa Công nghệ thông tin, Trường Đại học Công nghiệp thực phẩm TP. HCM

²Khoa Vật lý, Trường Đại học Sư phạm TP. HCM

³Phòng Quản lý khoa học và Đào tạo sau Đại học, Trường Đại học Công nghiệp thực phẩm TP. HCM

trangnthuy@hufi.edu.vn, thinhnv@hufi.edu.vn, tungnv@hufi.edu.vn, thanhvt@hufi.edu.vn, hainm@hcmue.edu.vn

TÓM TẮT: Trong bài báo này, chúng tôi tiếp cận một phương pháp phân cụm dữ liệu theo phân hoạch trên một cấu trúc cây đa nhánh để áp dụng tìm kiếm ảnh tương tự theo nội dung, cấu trúc cây này được gọi là GP-Tree. Mỗi phần tử dữ liệu được biểu diễn dưới dạng một vectơ mô tả đặc trưng của hình ảnh và được lưu trữ tại nút lá của cây GP-Tree. Việc tăng trưởng của cây dựa trên việc tách nút lá để tạo thành nhánh con mới nếu như số lượng phần tử tại nút đó lớn hơn một giá trị M cho trước. Vì vậy, cây GP-Tree tăng trưởng theo hướng từ nút gốc đến nút lá và tạo thành một cấu trúc cây phân cấp với các nút trong của cây chứa các phần tử đại diện cho các nút con ở mức tiếp theo. Để xây dựng cấu trúc cây này, chúng tôi thực hiện các phép toán bao gồm: thêm phần tử, tách nút lá, cập nhật các phần tử đại diện từ lá đến gốc, tạo nút con và tạo nút đồng cấp. Từ đó, chúng tôi đưa ra mô hình và phương pháp tìm kiếm ảnh theo nội dung ứng với một ảnh truy vấn cho trước. Thực nghiệm được đánh giá hiệu suất trên các bộ ảnh COREL, đồng thời so sánh kết quả với các công trình khác gần đây. Theo như kết quả thực nghiệm, cấu trúc cây GP-Tree và mô hình truy vấn ảnh của chúng tôi là hiệu quả và có thể áp dụng cho các hệ truy vấn ảnh trong các lĩnh vực khác nhau.

Từ khóa: Truy vấn theo nội dung (CBIR), GP-Tree, gom cụm (clustering), ảnh tương tự (similar image).

I. GIỚI THIỆU

Trong những năm gần đây, nhiều công trình nghiên cứu về hệ thống truy vấn ảnh đã được công bố nhằm giải quyết bài toán tìm kiếm ảnh tương tự trong các hệ dữ liệu đa phương tiện như tìm kiếm ảnh dựa trên thuật toán K-Means [13] hệ truy vấn theo nội dung QBIC [6], Virage [19], Photobook [8], VisualSEEK [4], Netra [21], SIMPLIcity [24], tìm kiếm ảnh dựa trên hình dạng, màu sắc, cấu trúc, đối tượng đặc trưng [10, 11]. Truy vấn ảnh tương tự được áp dụng trong nhiều lĩnh vực khác nhau như y tế, thời trang, hệ thống giám sát đối tượng, hệ thống thông tin địa lý, thư viện số... [2, 7].

Các công trình đã khảo sát tập trung vào kỹ thuật trích xuất đặc trưng, kỹ thuật đối sánh và tìm kiếm dựa trên các đặc trưng, dẫn đến tốn kém nhiều chi phí về thời gian và bộ nhớ khi đối sánh hai đối tượng hình ảnh. Do đó, cần có một phương pháp tra cứu hình ảnh tương tự dựa trên một cấu trúc dữ liệu trung gian để tăng hiệu suất truy vấn. Mặt khác, việc tìm kiếm dữ liệu trung gian cần sử dụng các phương pháp khai phá dữ liệu để tìm ra tập dữ liệu đại diện cho hình ảnh. Trong đó, phương pháp phân cụm là một trong những kỹ thuật quan trọng trong khai thác dữ liệu và đã được ứng dụng trong nhiều hệ thống tìm kiếm ảnh đã được phát triển cho các cơ sở dữ liệu lớn, một thuật toán tiêu biểu trong phân cụm phân hoạch là K-Means [15].

Tuy nhiên, kết quả phân cụm thu được từ thuật toán K-Means phụ thuộc vào việc khởi tạo số lượng cụm ban đầu, điều này ảnh hưởng đến độ chính xác của quá trình phân cụm, nghĩa là phụ thuộc vào số lượng tâm cụm đã được chọn ban đầu. Ngoài ra, nếu bổ sung phần tử mới vào cụm thì thuật toán K-Means cần phải được xác định lại tâm cụm mới, điều này làm cho tốn kém nhiều chi phí trong quá trình thực thi. Bên cạnh đó, nếu dữ liệu tăng trưởng ngày càng lớn thì việc xác định trước số lượng tâm cụm ban đầu là không phù hợp bởi vì có thể dẫn đến hai phần tử trong cùng một cụm có khoảng cách khá lớn [22]. Do đó, trong bài báo này, chúng tôi tiếp cận một cấu trúc phân cụm phân hoạch dựa trên một bán kính θ cho trước. Vì số lượng phần tử dữ liệu tăng trưởng nên một số cụm được giãn nở cho đến khi số lượng phần tử lớn hơn M cho trước để từ đó thao tác tách cụm được thực hiện và tạo thành một cấu trúc cây phân cấp, được gọi là GP-Tree.

Trong bài báo này, chúng tôi sử dụng một giá trị ngưỡng θ để đánh giá độ tương tự giữa các đối tượng dữ liệu. Khi dữ liệu tăng trưởng theo sự gia tăng của số lượng hình ảnh thì chúng tôi không cần mất nhiều thời gian trong việc phân cụm lại. Trên cơ sở này, chúng tôi xây dựng cấu trúc phân cụm phân hoạch dựa trên một bán kính θ cho trước. Chúng tôi đã ứng dụng thực nghiệm trên bộ ảnh COREL để đánh giá độ chính xác và tính khả thi cho những đề xuất.

Đóng góp của bài báo bao gồm: (1) Xây dựng một cấu trúc cây phân hoạch phân cấp GP-Tree nhằm lưu trữ dữ liệu tại các nút lá của cây; (2) Quy tắc tạo cụm mới dựa trên ngưỡng θ nhằm phù hợp với bộ dữ liệu tăng trưởng; (3) Đề xuất các thuật tách cụm khi số lượng phần tử của cụm lớn hơn M (M là hằng số quy định); (4) Thuật toán tìm kiếm ảnh tương tự theo nội dung dựa vào cấu trúc GP-Tree; (5) Xây dựng mô hình thực nghiệm và thực thi trên một bộ dữ liệu phổ dụng nhằm minh chứng tính đúng đắn của lý thuyết đề xuất.

Phần tiếp theo của bài báo này như sau: Phần II, đề cập đến các công trình liên quan nhằm phân tích, đánh giá tính khả thi của phương pháp tiếp cận bài báo; Phần III, trình bày chi tiết về kỹ thuật tạo cấu trúc cây phân cấp (GP-

Tree); Phần IV, mô tả thực nghiệm và đánh giá hiệu quả của phương pháp đề xuất nhằm chứng minh tính đúng đắn của cơ sở lý thuyết; Kết luận và hướng phát triển được trình bày ở Phần V.

II. CÁC CÔNG TRÌNH LIÊN QUAN

Trong những năm gần đây, có nhiều nhóm nghiên cứu về nâng cao hiệu quả cho bài toán tìm kiếm ảnh như tìm kiếm ảnh dựa trên phương pháp đánh trọng số đặc trưng vùng trên ảnh và ứng dụng kỹ thuật học không giám sát [6], truy vấn ảnh dựa trên kỹ thuật phân lớp bằng máy vectơ hỗ trợ SVM [8], truy vấn ảnh dựa trên cấu trúc dữ liệu cây S-Tree [23], truy vấn ảnh theo ngữ nghĩa dựa trên mô hình phân lớp phân cấp bằng kỹ thuật mạng nơ-ron học sâu [7], truy vấn ảnh theo nội dung sử dụng kỹ thuật máy vectơ hỗ trợ SVM [18]...

Năm 2015, A. Alzu'bi và cộng sự khảo sát và đánh giá tính khả thi của hệ truy vấn ảnh theo nội dung CBIR. Trong công trình này, nhóm tác giả đã phân tích những kỹ thuật cơ sở cho bài toán CBIR đồng thời đề xuất các đặc trưng của hình ảnh cần phải có để thực hiện quá trình truy vấn ảnh. Các kỹ thuật về máy học, khai phá dữ liệu cũng được giới thiệu để làm cơ sở xây dựng bài toán truy vấn ảnh. Theo đó, nhóm tác giả đã giới thiệu nhiều bộ dữ liệu có thể được ứng dụng để xây dựng các hệ thống truy vấn ảnh nhằm đánh giá tính hiệu quả của từng phương pháp đề xuất. Từ công trình này cho thấy bài toán truy vấn ảnh theo nội dung và theo ngữ nghĩa là một bài toán khả thi và có thể áp dụng cho nhiều hệ thống đa phương tiện khác nhau [1].

Năm 2016, A. L. Mane xây dựng hệ truy vấn ảnh bằng cách trích xuất đặc trưng về nội dung của hình ảnh đồng thời đánh chỉ mục. Nhóm tác giả đã sử dụng bộ ảnh COREL để đánh giá kết quả thực nghiệm cũng như tính hiệu quả của phương pháp đề xuất. Tuy nhiên, trong công trình này tác giả thực hiện việc đánh chỉ mục ngữ nghĩa trên cơ sở chủ quan, không có một mô hình phân loại tự động nhằm ứng dụng tổng quát cho nhiều bộ ảnh khác nhau [13].

Kusrini và nhóm tác giả (2016) [17] đề xuất một tiếp cận tìm kiếm ảnh theo nội dung dựa trên dựa trên cấu trúc cây quyết định kết hợp với thuật toán phân cụm K-Means dựa trên các đặc trưng về màu sắc và độ tương phản. Tuy nhiên, việc phân cụm tập ảnh dựa trên độ đo Euclide giữa các ảnh theo đặc trưng màu sắc và độ tương phản chưa tốt dẫn đến hệ thống truy vấn tập ảnh tương tự ảnh truy vấn đạt hiệu suất chưa cao. Vijayarajan và cộng sự (2016) [30] đã thực hiện tra cứu ảnh dựa trên ngữ nghĩa của ảnh. Tuy nhiên, phương pháp này chưa thực hiện phân lớp nội dung hình ảnh từ các đặc trưng màu sắc và đặc trưng không gian để tạo các từ khóa nhằm thực hiện tra cứu; do đó chưa thực hiện quá trình tìm kiếm từ một hình ảnh truy vấn cho trước.

Năm 2017, Hakan Cevikalp và cộng sự đã thực hiện bài toán tìm kiếm ảnh trên các hình ảnh có kích thước lớn. Trong bài báo này, nhóm tác giả đã xây dựng cây phân cấp nhị phân và máy vectơ hỗ trợ để phân loại các lớp đối tượng thị giác của hình ảnh. Thực nghiệm của bài báo được ứng dụng trên bộ ảnh ImageCLEF để đánh giá độ chính xác của phương pháp đề xuất. Tuy nhiên, cấu trúc cây nhị phân trong bài báo này chỉ dùng để phân lớp các hình ảnh, vẫn chưa tạo một cấu trúc lưu trữ cho hình ảnh. Do đó, tốc độ truy xuất và tốc độ tìm kiếm của hình ảnh còn hạn chế. Mặt khác, trong công trình này chưa phân tích cũng như trích xuất ngữ nghĩa thị giác của ảnh truy vấn [8].

Wan và các cộng sự (2017) [25] đề xuất một tiếp cận tìm kiếm ảnh theo nội dung dựa trên CD-Tree. Các nút trong CD-Tree được đánh chỉ mục và sử dụng thuật toán quay lui một chiều để tăng hiệu suất truy vấn tập ảnh tương tự cho ảnh truy vấn. Tuy nhiên, tốc độ truy vấn trên cây chưa tốt do giải thuật quay lui trên toàn bộ các nhánh của cây. Cũng trong năm này, Allani Olfa và cộng sự [29] đề xuất hệ thống truy xuất hình ảnh SemVisIR, kết hợp các tính năng cấp thấp của hình ảnh và ngữ nghĩa cấp cao. Bộ dữ liệu hình ảnh được lưu trữ trong một biểu đồ mẫu được tạo tự động bằng các thuật toán phân cụm. SemVisIR đã mô hình hóa các khía cạnh trực quan của hình ảnh thông qua các biểu đồ vùng và gán chúng cho các mô đun Ontology được xây dựng tự động. Hệ thống được triển khai và đánh giá trên bộ dữ liệu ImageCLEF.

Ouiem Bchir và cộng sự (2018) [26] đã thực hiện truy vấn ảnh dựa trên việc trích xuất vectơ đặc trưng của đối tượng vùng để thực hiện quá trình phân hoạch nhằm tăng tốc độ tìm kiếm ảnh. Trong phương pháp này, nhóm tác giả xây dựng một ánh xạ ngữ nghĩa giữa đặc trưng thị giác và ngữ nghĩa cấp cao. Nhóm tác giả đã thực nghiệm trên bộ ảnh ImageCLEF và cho thấy tính hiệu quả của phương pháp đề xuất. Tuy nhiên, trong công trình này chưa tạo ra một mô hình tìm kiếm trên một cấu trúc phân cấp nhằm tăng tính hiệu quả về thời gian và độ chính xác [6].

Fahd A.Alqasemi và các cộng sự (2019) [27] đề xuất một tiếp cận tìm kiếm ảnh theo nội dung dựa trên việc thống kê các đặc trưng về màu sắc của hình ảnh trong không gian RGB; phương pháp kết hợp với kỹ thuật k-NN với độ đo tương đồng Euclide để xác định tập ảnh tương tự. Tuy nhiên, phương pháp đề xuất chỉ mới thực hiện việc tìm kiếm tập ảnh tương tự theo tuyến tính, chưa tổ chức một cấu trúc dữ liệu để xác định tập ảnh láng giềng cho bài toán tìm tập ảnh tương tự.

Theo Shuang Jia và cộng sự (2020) [28], đề xuất một tiếp cận tìm kiếm ảnh theo nội dung dựa trên thuật toán gom cụm K-Means và tui từ thị giác. Hệ thống đề xuất được thực hiện bằng cách gom cụm các ảnh dựa trên thuật toán K-Means, sau đó trích xuất tập vectơ từ thị giác của các cụm. Với mỗi ảnh truy vấn, hệ thống sẽ xác định cụm có tui từ xuất hiện nhiều từ của ảnh truy vấn nhất; sau đó tiến hành trích xuất tập ảnh tương tự cho ảnh truy vấn. Tuy nhiên, trong phương pháp này các tui từ là độc lập và chưa phân lớp được nội dung của hình ảnh, nên việc truy vấn một ảnh bất kỳ chưa gán nhãn sẽ đạt độ chính xác không cao.

Dựa trên các công trình nghiên cứu đã phân tích trên, phương pháp được đề xuất trong chuyên đề này được thực hiện như sau: với mỗi bộ dữ liệu hình ảnh được trích xuất các đặc trưng cấp thấp để từ đó làm cơ sở thực hiện khai phá dữ liệu cho hình ảnh; kết hợp giữa kỹ thuật gom cụm phân hoạch K-Means và cấu trúc dữ liệu cây phân cụm phân cấp, được gọi là GP-Tree; đồng thời thực hiện phân lớp k-NN (k-Nearest Neighbor) nhằm tìm kiếm tập ảnh tương tự của một ảnh truy vấn đầu vào.

III. CÂY PHÂN CẤP GP-TREE

3. Cây phân cụm GP-TREE

3.1. Cấu trúc cây GP-Tree

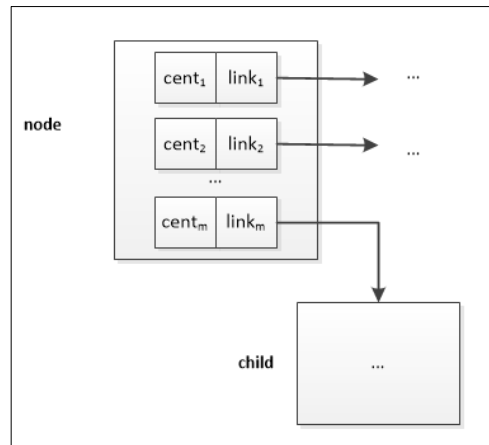
Trên cơ sở cấu trúc cây đa nhánh và phương pháp phân cụm K-Means, cấu trúc cây GP-Tree được xây dựng bằng cách tách nút lá thành hai nút nếu như số lượng phần tử tại nút lá đó vượt trên giá trị M cho trước, đồng thời tại mỗi một nút trong các nút con có thể được tạo thành dựa trên một ngưỡng θ , ngưỡng này nhằm đánh giá độ tương tự của dữ liệu. Nghĩa là, nếu một phần tử dữ liệu có độ sai biệt tương tự lớn hơn ngưỡng θ thì phần tử dữ liệu đó phải thuộc về một nhánh khác. Do đó, cây GP-Tree tăng trưởng theo hướng lá và phát triển thành cây đa nhánh mà mỗi một nút lá là một cụm dữ liệu gồm các phần tử tương tự nhau. Trong bài báo này, các phần tử dữ liệu là các vectơ đặc trưng cho mỗi hình ảnh và lần lượt được lưu trữ trên cây GP-Tree, từ đó các thao tác trên cây được thực hiện.

GP-Tree là một cây đa nhánh gồm một nút gốc, một tập các nút trong và một tập các nút lá. Mỗi nút lá trong cây GP-Tree chứa cụm các ảnh tương tự. Việc tạo GP -Tree dựa trên thao tác: tách nút, thêm phần tử, tạo nhánh con mới. Quá trình tìm kiếm ảnh được thực hiện bằng cách duyệt từ nút gốc và chọn một nhánh nếu như phần tử đại diện của nhánh đó có độ tương tự gần nhất với dữ liệu ảnh truy vấn. Quá trình này được thực hiện tương tự nhưng với phần tử nút con kế tiếp nếu như chưa phải là nút lá; ngược lại thì nút lá đó chứa tập ảnh tương tự với ảnh truy vấn.

Để xây dựng cây GP-Tree, các phần tử dữ liệu ed tại nút lá và phần tử đại diện $cent$ tại nút trong (nút gốc) được xây dựng dưới dạng các vectơ có n -chiều và được định nghĩa như sau:

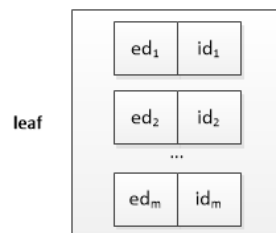
Định nghĩa 1. Phần tử dữ liệu ed và $cent$ là các vectơ có n -chiều sao cho:

- (1) Mỗi phần tử ed là một vectơ (f_1, f_2, \dots, f_n) với $f_i \in [0,1]$
- (2) Mỗi phần tử $cent$ là một vectơ (c_1, c_2, \dots, c_n) với $c_i \in [0,1]$ và $c_i = \text{Average}\{\text{child}, \text{cent}_k\}$, với $k = 1..K$



Hình 1. Một cấu trúc nút trong của cây GP-Tree

Trong Hình 1, các phần tử đại diện của cụm nút con và các liên kết đến nút con kế tiếp được mô tả dựa trên mỗi cặp phần tử $\langle cent, link \rangle$. Trong đó, mỗi $cent$ là một vectơ gồm các giá trị là số thực nhằm định hướng đường đi trong quá trình tìm kiếm nút lá phù hợp với mỗi vectơ hình ảnh đầu vào; Mỗi $link$ liên kết tới nút con kế tiếp nhằm xác định vị trí nút lá cần tìm.



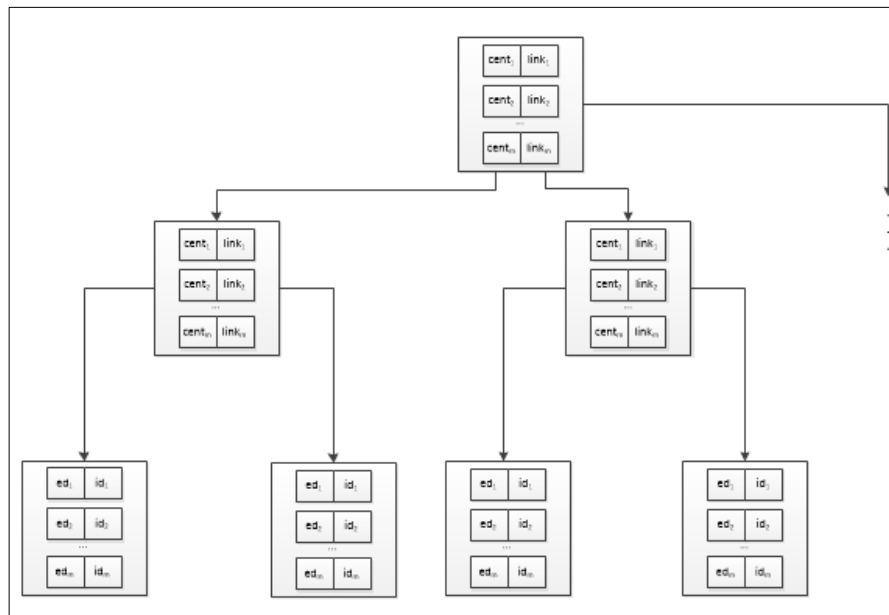
Hình 2. Một cấu trúc nút lá trong cây GP-Tree

Hình 2 mô tả một cấu trúc nút lá bao gồm một danh sách các phần tử dữ liệu, sao cho mỗi phần tử dữ liệu là một cặp giá trị $\langle ed, id \rangle$ với ed là véctơ đặc trưng của hình ảnh và id là định danh liên kết với hình ảnh thực sự. Cặp phần tử $\langle ed, id \rangle$ chỉ xuất hiện trong nút lá, do đó, tại nút lá không có trường liên kết đến nút tiếp theo.

Gọi $cent_i, ed_i, link_i, id_i$ lần lượt là phần tử đại diện, phần tử tại nút lá, liên kết đến nút con kế tiếp và định danh hình ảnh của phần tử dữ liệu thứ i tại một nút trong cây; m, N, L lần lượt là số lượng phần tử tại nút hiện hành, số lượng các nút trong và số lượng các nút lá của cây. Khi đó, cây GP-Tree được định nghĩa như sau:

Định nghĩa 2. Cây GP-Tree là cây đa nhánh bao gồm:

- (1) Một nút gốc $root$ chứa một tập các phần tử $C = \{\langle cent_i, link_i \rangle \mid i = 1..m\}$
- (2) Một tập nút trong $\mathcal{N} = \{\mu_k \mid k = 1..N\}$, với $\mu_k = \{\langle cent_i^k, link_i^k \rangle \mid i = 1..m\}, link_i^k \neq null$;
- (3) Một tập nút lá $\mathcal{E} = \{\mu_l \mid l = 1..L\}$, với $\mu_l = \{\langle ed_i^l, id_i^l \rangle \mid i = 1..m\}$



Hình 3. Một cấu trúc cây GP-Tree bao gồm nút gốc, các nút trong và các nút lá

Cấu trúc cây GP-Tree được mô tả trong Hình 3 bao gồm một nút gốc liên kết đến các nút trong; Nút gốc và nút trong lưu trữ các phần tử đại diện và liên kết đến các nút con kế tiếp. Mỗi nút lá có một con đường đi từ nút gốc dựa trên các phần tử đại diện và các liên kết tại các nút trong. Do đó, quá trình tìm kiếm một nút lá ứng với ảnh truy vấn đầu vào được dựa trên quá trình định hướng tại các nút trong. Trên cơ sở này, tập các ảnh tương tự sẽ được trích xuất dựa trên các định danh hình ảnh tại nút lá.

3.2. Các thao tác xử lý trên cây GP-Tree

a. Cập nhật các nút đại diện

Khi thêm các phần tử dữ liệu $\langle ed, id \rangle$ vào cây GP-Tree thì số phần tử tại nút lá thay đổi dẫn đến phần tử đại diện bị thay đổi theo. Do đó, các phần tử đại diện này phải được cập nhật từ nút lá tới nút gốc. Trong trường hợp phát sinh một phần tử của nút trong thì phần tử đại diện của nút đó cũng phải được cập nhật đến nút gốc. Thuật toán cập nhật phần tử đại diện được mô tả như sau:

Thuật toán **UPCE**

Đầu vào: Một nút *Node* (có thể là lá hoặc nút trong), cây GP-Tree

Đầu ra: cây GP-Tree sau cập nhật

Begin

If(Node = Root) **then**
Return GP – Tree

Else

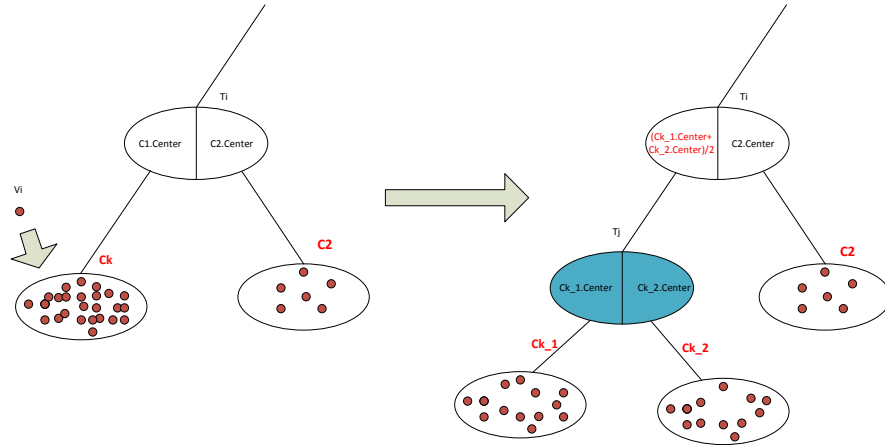
cent = Average {element_i | element_i ∈ Node}
Node.Parent.cent_{Node} = cent;
Node = Node.Parent;
UPCE(Node, GP – Tree);

EndIf

End

b. Tách một nút lá

Gọi M là số phần tử tối đa tại nút lá, nếu số phần tử tại một nút lá bất kỳ lớn hơn M thì nút lá này được tách thành hai nút lá đồng thời tạo ra một nút cha liên kết tới hai nút này và nút cha trở thành nút con của nút cha hiện tại. Hình 4 mô tả phương pháp tách nút để tạo thành một nút trong mới liên kết đến hai nút lá sau khi tách nút lá ban đầu.



Hình 4. Mô tả phương pháp tách nút lá trên cây GP-Tree

Quá trình tách nút lá trên cây GP-Tree được mô tả như sau:

- (1) Tìm véctor v_e và v_f cách xa nhau nhất trong cụm C_k
- (2) Tạo cụm C_{k_1} có tâm v_e ; cụm C_{k_2} có tâm v_f
- (3) Lấy $v_i \in C_k$: nếu $d(v_i, C_{k_1}.center) \leq d(v_i, C_{k_2}.center)$ thì $v_i \in C_{k_1}$. Ngược lại $v_i \in C_{k_2}$.

Trên cơ sở các đề xuất như trên thuật toán tách nút lá trên cây GP-Tree được mô tả như sau:

Thuật toán SPLN

Đầu vào: Cây GP-Tree, nút lá L

Đầu ra: Cây GP-Tree sau khi tách

Begin

Tạo 2 nút lá rỗng: $L_s = null$ và $L_t = null$

Tìm ed_s, ed_t : $Euclide(ed_s, ed_t) = \max\{Euclide(ed_i, ed_j) \mid i, j = 1..m\}$

$L_s = L_s \cup ed_s$

$L_t = L_t \cup ed_t$

ForEach $ed_i \in L$ do

If ($Euclide(ed_i, ed_s) \leq Euclide(ed_i, ed_t)$)

$L_s = L_s \cup ed_i$

Else

$L_t = L_t \cup ed_i$

EndIf

EndForEach

Tạo hai phần tử đại diện cho hai nút lá L_s và L_t

$cent_s = Average\{ed_i\}; ed_i \in L_s$

$cent_t = Average\{ed_i\}; ed_i \in L_t$

Tạo một nút $Node\{\langle cent_s, link_s \rangle, \langle cent_t, link_t \rangle\}$

$Node.link_s = L_s$

$Node.link_t = L_t$

Tạo liên kết nút cha

$cent_{Node} = Average(cent_s, cent_t)$

$L.Parent = L.Parent \cup cent_{Node}$

$L.Parent.cent_{Node}.link = Node$

UPCE($L.Parent, GP-Tree$)

Return GP-Tree;

End

c. Thêm một phần tử vào cây

Vì cây GP-Tree là cây tăng trưởng bằng cách thêm một phần tử vào nút lá dựa trên một ngưỡng θ cho trước nên độ đo tương tự d giữa phần tử cần thêm với phần tử đại diện có các trường hợp xảy ra như sau:

- (1) Nếu $d \leq \theta$ thì phần tử này thuộc nút lá hiện tại.
- (2) Nếu $d > \theta$ thì tạo một nhánh mới tại nút cha hiện hành và liên kết đến nút lá mới.

Thuật toán thêm một phần tử vào cây GP-Tree được mô tả như sau:

Thuật toán INSE

Đầu vào: Cây GP-Tree, phần tử dữ liệu $\langle ed, id \rangle$, ngưỡng θ

Đầu ra: Cây GP-Tree

Begin

If (*root is leaf*) **then**

$root = root \cup \langle ed, id \rangle$

If(*root.count* > M) **then**

SPLN ($GP - Tree, Root$)

EndIf

Return $GP - Tree$

EndIf

Tìm $Node_m$: $Euclide(ed, cent_m) = \min\{Euclide(ed, cent_i) | cent_i \in root\}$

If($Euclide(ed, cent_m) \leq \theta$) **then**

Gọi đệ quy INSE($GP - Tree, Node_m, \langle ed, id \rangle, \theta$)

Else

Tạo nhánh mới

$Node = Node \cup \langle ed, link_m \rangle$

$Leaf = \{\langle ed, id \rangle\}$

$Node.Link_m = Leaf$

UPCE($Node, GP - Tree$)

EndIf

End

d. Tạo cây GP-Tree

Việc tạo cây được thực hiện bằng cách thêm từng phần tử $\langle ed, id \rangle$ vào trong cây GP-Tree. Quá trình thêm này sẽ thực hiện việc tách nút và cây sẽ tăng trưởng để chứa bộ dữ liệu ban đầu. Nếu sau khi thực hiện tạo cây cho bộ dữ liệu mà dữ liệu vẫn tiếp tục tăng trưởng thì việc thêm phần tử này được thực hiện dựa vào thuật toán **INSE** và thuật toán **UPCE**. Vì vậy, quá trình tạo cây được mô tả qua thuật toán sau đây.

Thuật toán GCGP_TREE

Đầu vào: Tập dữ liệu ảnh $IMGDB$, ngưỡng θ

Đầu ra: Cây GP-Tree

Begin

$GP - Tree = null$

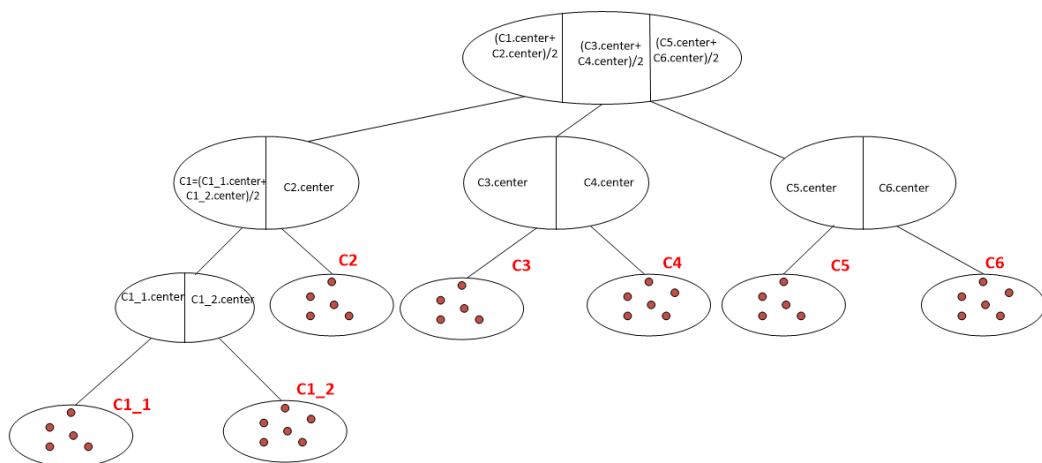
Foreach($\langle ed, id \rangle \in IMGDB$) **do**

$GP - Tree = INSE(\langle ed, id \rangle, GP - Tree);$

EndForeach

Return GP-Tree

End



Hình 5. Một ví dụ về cấu trúc trên cây GP-Tree

d. Tìm kiếm ảnh tương tự trên cây GP-Tree

Với mỗi véctơ đặc trưng ed của ảnh truy vấn được so sánh với các phần tử đại diện tại các nút trong cây từ gốc đến lá. Tại mỗi nút trong cây thuật toán sẽ định hướng đến nhánh gần nhất với phần tử ed theo độ đo Euclide, cho đến khi tìm được nút lá phù hợp. Các phần tử tại nút lá này là các định danh của ảnh tương tự với ảnh truy vấn đầu vào. Thuật toán tìm kiếm trên cây GP-Tree được mô tả như sau:

Thuật toán **Retrieval_IRGP**

Đầu vào: $root$, véctơ đặc trưng ed

Đầu ra: tập Ψ chứa các id (định danh) của các ảnh tương tự.

Begin

If($root$ is leaf) **then**

$\Psi = \{ \langle ed_i, id_i \rangle \in root \}$

Else

Tìm hướng đi ngắn nhất $cent_m$: $Euclide(ed, cent_m) = \min\{ Euclide(ed, cent_i) \}$ với $cent_i \in root$

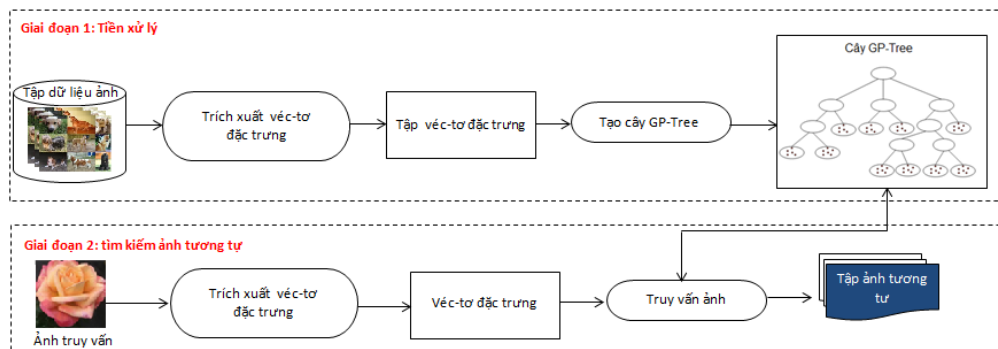
Gọi đệ quy hàm $Retrieval_IRGP(root.link_m, ed)$

EndIf

Return Ψ ;

End

IV. MÔ HÌNH TRUY VẤN ẢNH



Hình 6. Mô hình truy vấn ảnh theo nội dung dựa trên cây phân cấp GP-Tree

Mô hình thực nghiệm của hệ thống tìm kiếm ảnh theo nội dung dựa trên cây GP-Tree được mô tả tại Hình 6. Mô hình tìm kiếm ảnh gồm hai pha: tiền xử lý và tìm kiếm ảnh tương tự.

A. Pha tiền xử lý

Bước 1: Phân đoạn và trích xuất véctơ đặc trưng của tập dữ liệu ảnh.

Bước 2: Tạo cây phân cụm phân cấp GP-Tree.

B. Pha tìm kiếm ảnh tương tự

Bước 1: Trích xuất véctơ đặc trưng ảnh cần truy vấn.

Bước 2: Truy vấn véctơ đặc trưng ảnh cần truy vấn trong cây GP-tree để trích được tập ảnh tương tự.

Bước 3: Kết xuất các ảnh kết quả sắp xếp theo độ đo tương tự với ảnh truy vấn.

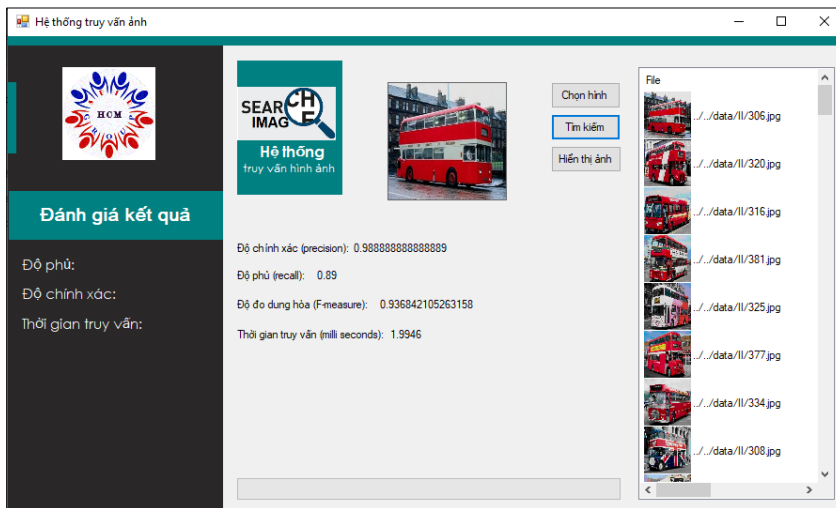
V. THỰC NGHIỆM

5.1. Môi trường thực nghiệm

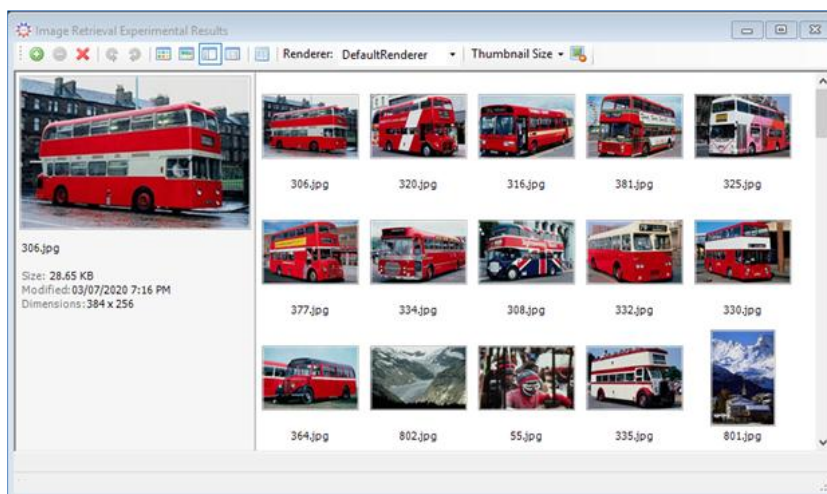
Thực nghiệm được xây dựng gồm hai giai đoạn: (1) Giai đoạn tiền xử lý nhằm tạo ra tập véctơ đặc trưng cho tập dữ liệu hình ảnh, tạo cây phân cụm phân cấp GP-tree trên tập các véctơ dựa trên thuật toán đã được đề xuất; (2) tìm kiếm tập ảnh tương tự. Tất cả các ứng dụng thực nghiệm được xây dựng trên nền tảng dotNET Framework 3.5, ngôn ngữ lập trình C#. Thực nghiệm trên máy PC CPU Intel Core i3-7100U CPU @2.40GHz, 8.0 GB RAM, hệ điều hành Windows 10 Pro 64 bit.

5.2. Ứng dụng thực nghiệm

Quá trình thực nghiệm được truy vấn trên các bộ ảnh COREL, Wang, ImageCLEF. Với mỗi hình ảnh truy vấn sẽ tìm ra các hình ảnh có độ tương tự nhiều nhất với hình ảnh truy vấn, từ đó đối sánh với danh mục chủ đề hình ảnh nhằm đánh giá độ chính xác của phương pháp. Hệ thống giúp truy vấn ảnh, với mỗi hình ảnh truy vấn sẽ được trích lọc trên tập dữ liệu ảnh và tìm ra các hình ảnh có độ tương tự nhiều nhất với hình ảnh truy vấn.



Hình 7a. Một kết quả mẫu về truy vấn ảnh tương tự

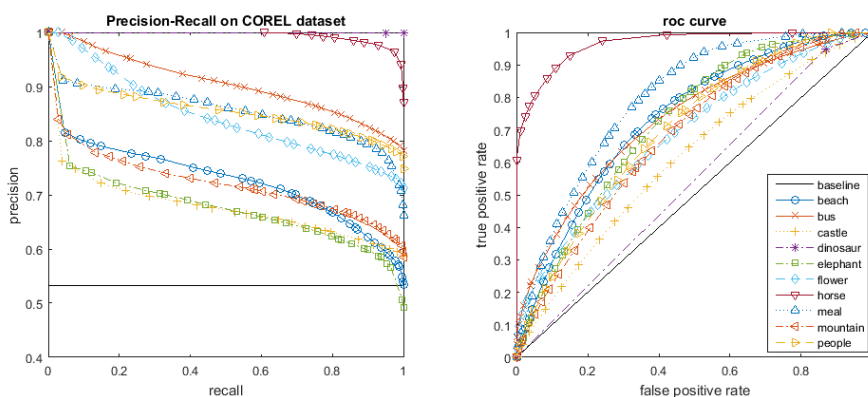


Hình 7b. Một kết quả mẫu về truy vấn ảnh tương tự

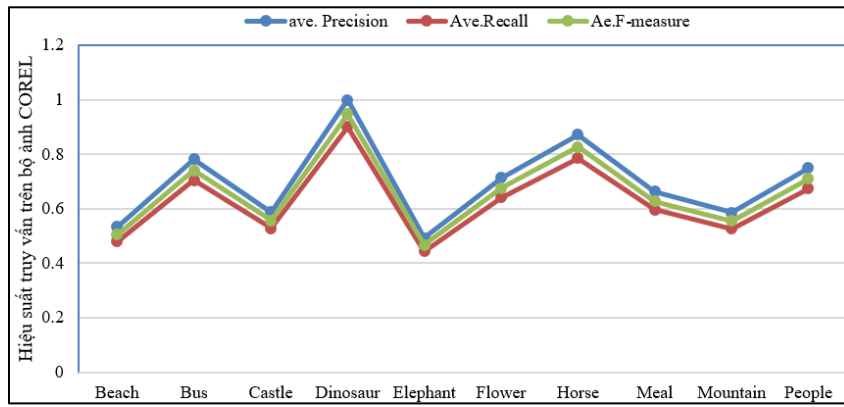
5.3. Kết quả thực nghiệm

Kết quả thực nghiệm đối với phương pháp tìm kiếm ảnh được thể hiện trên đồ thị mô tả hiệu suất truy vấn trên bộ ảnh COREL được minh họa, mô tả độ chính xác (*precision*), độ phủ (*recall*) trong Hình 8 tương ứng với từng tập dữ liệu ảnh theo phân lớp. Mỗi đường cong của Hình 8 mô tả độ chính xác và độ phủ của từng ảnh trong mỗi chủ đề thuộc tập dữ liệu ảnh.

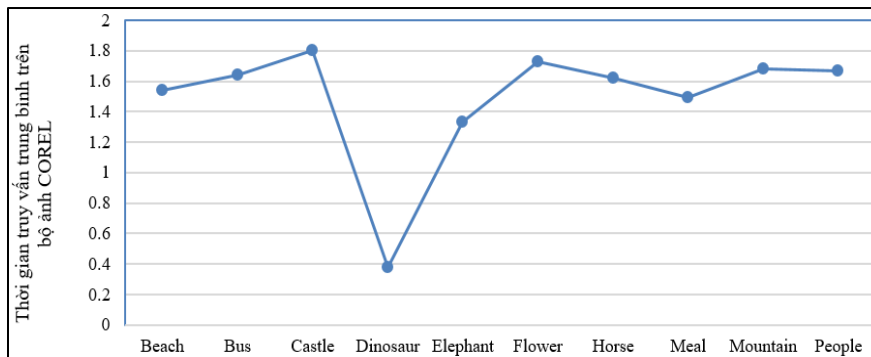
Hiệu suất truy vấn trên các phân lớp được mô tả tại Hình 9, thời gian truy vấn trung bình (*theo milli giây*) trên bộ ảnh COREL được mô tả trong Hình 10. Trong đó, mỗi phân lớp ảnh được thực hiện việc tìm kiếm ảnh tương tự theo mỗi ảnh đồng thời chúng tôi tính thời gian tìm kiếm và giá trị trung bình của độ chính xác.



Hình 8. Đồ thị mô tả hiệu suất truy vấn trên bộ ảnh COREL



Hình 9. Đồ thị mô tả hiệu suất truy vấn trên các phân lớp của bộ ảnh COREL



Hình 10. Thời gian truy vấn trung bình (theo milli giây) trên bộ ảnh COREL

Các giá trị về hiệu suất, thời gian tìm kiếm theo từng chủ đề và đánh giá so sánh cũng được trình bày cụ thể trong Bảng 2, Bảng 3 và Bảng 4. Theo như kết quả thực nghiệm trong các bảng này, phương pháp đề xuất của chúng tôi cho bài toán tìm kiếm ảnh tương tự là hiệu quả, trong đó độ chính xác thấp nhất là 49,27 %, độ chính xác cao nhất là 100 %, độ chính xác trung bình là 69,84 %. Điều này cho thấy các ảnh đối tượng được truy vấn tốt hơn so với ảnh thiên nhiên phong cảnh.

Bảng 1. Bảng mô tả hiệu suất truy vấn trên các phân lớp của bộ ảnh COREL

Phân lớp ảnh	Ave. Precision	Ave. Recall	Ave.F-measure
Beach	0.533333333	0.48	0.505263158
Bus	0.782222222	0.704	0.741052632
Castle	0.586888889	0.5282	0.556
Dinosaur	1	0.9	0.947368421
Elephant	0.492666667	0.4434	0.466736842
Flower	0.713222222	0.6419	0.675684211
Horse	0.871888889	0.7847	0.826
Meal	0.662222222	0.596	0.627368421
Mountain	0.584888889	0.5264	0.554105263
People	0.748444444	0.6736	0.709052632

Bảng 2. Bảng mô tả thời gian truy vấn trung bình trên bộ ảnh COREL

Phân lớp ảnh	Ave.time (milli seconds)
Beach	1.540088
Bus	1.640091
Castle	1.800112
Dinosaur	0.380025
Elephant	1.372
Flower	1.7301
Horse	1.620093
Meal	1.490086
Mountain	1.680094
People	1.670093

Bảng 3. Giá trị hiệu suất, thời gian truy vấn trung bình trên bộ ảnh COREL

Ave. Precision	Ave.Recall	Ave.F-measure	Ave.Time (milli seconds)
0.69843466	0.6282526	0.661008405	1.485668733

Để minh chứng cho mô hình truy vấn ảnh theo nội dung đề xuất là hiệu quả, chúng tôi so sánh kết quả thực nghiệm với một số công trình gần đây trên cùng bộ dữ liệu trong Bảng 4.

Bảng 4. So sánh hiệu suất truy vấn giữa các phương pháp trên bộ dữ liệu COREL

Phương pháp	Ave. Precision
Relevance Feedback, 2004 [23]	0.68
A. Huneiti, 2015 [24]	0.5588
Bella M. I. T., 2019 [25]	0.6090
Phương pháp đề xuất	0.6984346

Theo như kết quả phương pháp đề xuất, mô hình truy vấn ảnh của chúng tôi có tính khả thi và áp dụng hiệu quả trong các hệ truy vấn ảnh theo nội dung trên nhiều lĩnh vực khác nhau.

Kết quả thực nghiệm cho thấy, thuật toán của chúng tôi đã xây dựng được chương trình tìm kiếm ảnh hiệu quả, nghĩa là thời gian tìm kiếm nhanh và có độ chính xác cao. Điều này đã minh chứng tính đúng đắn theo thực nghiệm của các phương pháp đề xuất và đáp ứng được mục tiêu đề ra ban đầu bài báo. Từ đó cho thấy phương pháp tìm kiếm ảnh tương tự dựa vào cây GP-Tree của chúng tôi là hiệu quả và có thể áp dụng để xây dựng công cụ tìm kiếm ảnh đáp ứng yêu cầu người dùng.

VI. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi xây dựng một cấu trúc cây phân cụm phân cấp nhằm phân hoạch dữ liệu để từ đó áp dụng cho bài toán tìm kiếm ảnh tương tự. Các thuật toán tạo cây GP-Tree đã được đề xuất nhằm tạo ra một mô hình tăng trưởng dữ liệu và phân hoạch trên cây theo phương pháp học bán giám sát. Từ đó, mô hình và phương pháp tìm kiếm ảnh theo nội dung ứng với một ảnh truy vấn cho trước được đề xuất. Thực nghiệm được đánh giá hiệu suất trên các bộ ảnh COREL đồng thời so sánh kết quả với các công trình khác gần đây trên cùng một bộ dữ liệu. Theo như lý thuyết đề xuất và kết quả thực nghiệm, cấu trúc cây GP-Tree của chúng tôi là hiệu quả và có thể áp dụng cho các hệ truy vấn dữ liệu đa phương tiện. Hướng phát triển tiếp theo của bài báo này thực hiện kết nối các nút lá để tạo một đồ thị láng giềng tại tầng nút lá của cây GP-Tree, đồng thời mỗi phần tử tại nút lá được kết hợp với túi từ thị giác để phân lớp và trích xuất ngữ nghĩa cho mỗi hình ảnh dựa trên một ontology. Trên cơ sở này, các trọng số được đánh dấu trên các cung của đồ thị từ mỗi nút lá của cây để từ đó có thể tìm kiếm hiệu quả các hình ảnh tương tự về độ chính xác.

LỜI CẢM ƠN

Nhóm tác giả chân thành cảm ơn Trường Đại học Sư phạm TP. HCM, Trường Đại học Công nghiệp thực phẩm TP. HCM là những nơi hỗ trợ về chuyên môn và cơ sở vật chất cho nghiên cứu này. Chúng tôi trân trọng cảm ơn nhóm nghiên cứu SBIR-HCM đã hỗ trợ về chuyên môn giúp chúng tôi hoàn thành bài nghiên cứu này.

TÀI LIỆU THAM KHẢO

- [1] Anjali, T., N. Rakesh, and Krishna MP Akshay., "A novel based decision tree for content based image retrieval: an optimal classification approach.", International Conference on Communication and Signal Processing (ICCSP). IEEE, 2018.
- [2] Alzu'bi A, Amira A, Ramzan N, "Semantic content-based image retrieval: A comprehensive study", J. Vis. Commun. Image Rep., 32, pp. 20-54, 2015.
- [3] Amruta Laxman Mane, A.N. Bhute, "Semantic based image retrieval", Indian Journal of Computer Science and Engineering (IJCSSE), 2016.
- [4] J.R. Bach, C. Fuller, A. Gupta, et al., "Virage image search engine: an open framework for image management", in: SPIE 2670, 23, San Jose, CA, pp. 76-87, 1996.
- [5] Bella, M. I. T., & Vasuki, A., "An efficient image retrieval framework using fused information feature". Computers & Electrical Engineering, 75, pp. 46-60, 2019.
- [6] Ouiem Bchir, Mohamed Maher, Hadeel Aljam, "Region-Based image retrieval using relevance feature weights", International journal of fuzzy logic and intelligent systems international journal of fuzzy logic and intelligent systems Vol.18, No.1, pp. 65-77, 2018.

- [7] Björn Barz, Joachim Denzler, "Hierarchy-based Image Embeddings for Semantic Image Retrieval", IEEE Winter Conference on Applications of Computer Vision (WACV), number: 18493437, 2019.
- [8] H. Cevikalp, M. Elmas, S. Ozkan, "Large-scale image retrieval using transductive support vector machines", Computer Vision and Image Understanding, pp.1-11, 2017.
- [9] Dimitrovski, Ivica, et al., "Improving bag-of-visual-words image retrieval with predictive clustering trees". Information Sciences 329, pp. 851-865, 2016.
- [10] Gia, Giorgio, and Fabio Roli. "Instance-based relevance feedback for image retrieval", Advances in neural information processing systems. 2005.
- [11] Huneiti, A., & Daoud, M. "Content-based image retrieval using SOM and DWT". Journal of software Engineering and Applications, 8(02), pp. 51, 2015.
- [12] Fan, Xin-can., "Research on image semantic communication and retrieval based on massive resource library of internet", Journal of Computers 28.1, pp. 225-231, 2017.
- [13] Amruta Laxman Mane, A. N. Bhute, "Semantic based image retrieval", Indian Journal of Computer Science and Engineering (IJCSSE), 2016.
- [14] Y.-K. N. Hun-Woo Yoo, She-Hwan Jung, Dong-Sik Jang, "Extraction of major object features using VQ clustering for content-based image retrieval", Pattern Recognition 35, pp. 1115-1126, 2002.
- [15] R. Rakesh Kumar and A. Bhanu Prasad, "K- Means clustering algorithm for partitioning data sets evaluated from horizontal aggregations", e-ISSN: 2278-0661, p- ISSN: 2278-8727, Volume 12, Issue 5, pp. 45-48, 2013.
- [16] Khemchandani, Reshma, and Aman Pal., "Tree based multi-category Laplacian TWSVM for content based image retrieval", International Journal of Machine Learning and Cybernetics 8.4, pp. 1197-1210, 2017.
- [17] Kusrini, M. Dedi Iskandar, and Ferry Wahyu Wibowo., "Multi features content-Based image retrieval using clustering and decision tree algorithm", Telkomnika 14.4, pp. 1480, 2016.
- [18] Mehmood, Z., Mahmood, T., & Javid, M. A., "Content-based image retrieval and semantic automatic image annotation based on the weighted average of triangular histograms using support vector machine", Applied Intelligence, 48(1), pp. 166-181, 2017.
- [19] A. Pentland, R.W. Picard, S. Scarloff, "Photobook: tools for content-based manipulation of image databases", in: SPIE 2185, San Jose, CA, pp. 34-47, 1994.
- [20] [22] Salah, Marwa Ben, Ameni Yengui, and Mahmoud Neji., Study between Automatic Image Annotation Approaches.
- [21] J. R. Smith, "Integrated spatial and feature image systems: retrieval, analysis and compression", Ph.D. Dissertation, Columbia University, New York City, 1997.
- [22] Asmita Yadav and Sandeep Kumar Singh, "An Improved K-Means clustering algorithm", ISSN 2305-9184, Volume 5, Number 2, pp.88-103, 2016.
- [23] Văn Thế Thành, Lê Mạnh Thịnh, "Một số cải tiến cho hệ truy vấn ảnh dựa trên cây S-Tree", Kỹ yếu hội thảo quốc gia về nghiên cứu cơ bản và ứng dụng CNTT (FAIR), ĐH Cần Thơ, Nhà xuất bản Khoa học tự nhiên và Công nghệ, tr. 459-470, 2016.
- [24] J. Z. Wang, J. Li, G, "WiederholdSIMPLicity: semantics-sensitive integrated matching for picture libraries", IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 (9), pp. 947-963, 2001.
- [25] [20] Wan, Yuchai, Xiabi Liu, and Yi Wu., "Cd-tree: A clustering-based dynamic indexing and retrieval approach", Intelligent Data Analysis 21.2, pp. 243-261, 2017.
- [26] Ouiem Bchir, Mohamed Maher, Hadeel Aljam, "Region-based image retrieval using relevance feature weights" International journal of fuzzy logic and intelligent systems international journal of fuzzy logic and intelligent systems Vol.18, No.1, 2018.03, pp. 65-77, 2018.
- [27] Alqasemi, Fahd A., et al. "Feature Selection approach using KNN supervised learning for Content-Based Image Retrieval", First International Conference of Intelligent Computing and Engineering (ICOICE). IEEE, 2019.
- [28] Jia, Shuang, et al., "Bag-of-Visual Words based Improved Image Retrieval lgorithm for vision indoor positioning", 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). IEEE, 2020.
- [29] Olfa Allani, Hajer B. Z., et al., "Pattern graph-based image retrieval system combining semantic and visual features", Springer Science+Business Media New York, 2017.

- [30] M. D. V. Vijayarajan, P. Tejaswin, M. Lohani, "A generic framework for ontology-based information retrieval and image retrieval in web data", *Human-centric Computing and Information Sciences*, Vol. 6, No. 18, pp. 1-30, 2016.

A METHOD OF PARTITIONING DATA ON A HIERARCHICAL TREE FOR THE PROBLEM OF FINDING IMAGES BY CONTENT

Nguyen Thi Thuy Trang, Nguyen Minh Hai, Nguyen Van Tung, Nguyen Van Thinh, Van The Thanh

ABSTRACT: *In this paper, we approach an analytical data visualization approach to partitioning over a structured multi-branch tree, called GP-Tree. Each section data is represented as a vector descriptive feature of the image and is stored at the node leaf of the GP-Tree. Foudation of the tree based on the head of split node to create a new sub-branch if as the number of quality at that node that great more than a value for before. Thus, the GP-Tree grows in a direction from the node root to the node leaf and forms a structure tree level with the nodes in the container tree representing the elements for the next level nodes. To build this part structure, we perform math operations including: adding element, splitting the node leaf, updating the representative elements from leaf to root, creating node, and creating peer. From there, we give model and method of searching for images by application content with a given query image. The experiment was evaluated on the performance of COREL at the same time of the set of images to compare the results with other recent works. As far as the empirical results, our GP-Tree structure and query model, I am efficient and applicable to query systems in different domains.*

Keywords: *Query by content (CBIR), clustering, GP-Tree, similar image, ontology.*