Tạp chí Khoa học Đại học Huế: Kỹ thuật và Công nghệ; pISSN 2588-1175 | eISSN 2615-9732

Tập 131, Số 2A, 2022, Tr. 53–69; DOI: 10.26459/hueunijtt.v131i2A.6837

TÌM KIẾM ẢNH ĐA ĐỐI TƯỢNG DỰA TRÊN MẠNG R-CNN VÀ CẤU TRÚC KD-TREE

Đào Xuân Bao[3], Nguyễn Thị Định[1, 3], Nguyễn Phương Hạc[3], Văn Thế Thành[2]*

[1] Khoa Công nghệ Thông tin, Trường ĐH Khoa học, Đại học Huế, 77 Nguyễn Huệ, Huế, Việt Nam
[2] Trường Đại học Sư phạm Tp. HCM, 280 An Dương Vương, Quận 5, Tp. Hồ Chí Minh, Việt Nam
[3] Trường ĐH Công nghiệp Thực phẩm Tp. HCM, 140 Lê Trọng Tấn, Q. Tân Phú, Tp. Hồ Chí Minh, Việt Nam

Tóm tắt. Tìm kiếm ảnh đa đối tượng là một bài toán quan trọng trong lĩnh vực tra cứu ảnh do sự đa dạng và tính phức tạp của hình ảnh. Trong bài báo này, một phương pháp tìm kiếm ảnh đa đối tượng dựa trên mạng R-CNN và cấu trúc KD-Tree được đề xuất nhằm phát triển những ưu điểm của mạng R-CNN trong việc xác định và phân loại từng đối tượng riêng biệt trên ảnh, đồng thời kết hợp với cấu trúc KD-Tree trong việc lưu trữ hình ảnh đã mang lại hiệu suất truy vấn cao và thời gian tìm kiếm ổn định. Để giải quyết bài toán này, chúng tôi trích xuất và phân lớp các đối tượng trên tập dữ liệu hình ảnh bằng mô hình mạng R-CNN và lưu trữ trên cấu trúc KD-Tree. Từ đó, mỗi ảnh đầu vào được phân đoạn theo từng đối tượng, trích xuất vector đặc trưng và thực hiện tìm kiếm tập ảnh tương tự dựa trên cấu trúc KD-Tree. Trên cơ sở đó, một mô hình tìm kiếm ảnh dựa trên mạng R-CNN và cấu trúc KD-Tree được đề xuất. Để minh chứng cho tính đúng đắn của cơ sở lý thuyết đã đề xuất, thực nghiệm được xây dựng trên bộ ảnh COCO với hiệu suất tìm kiếm ảnh là 0,6898. Kết quả thực nghiệm được so sánh với các công trình khác cùng trên bộ dữ liệu; điều này minh chứng cho tính khả thi và hiệu quả của phương pháp đề xuất, đồng thời có thể ứng dụng cho các bộ ảnh đa đối tượng.

Từ khóa: R-CNN, KD-Tree, ảnh đa đối tượng, tìm kiếm ảnh, ảnh tương tự

Multi-object image retrieval based on R-CNN network and KD-Tree structure

Dao Xuan Bao[3], Nguyen Thi Dinh[1, 3], Nguyen Phuong Hac[3], Van The Thanh[2]*

[1] Faculty of Information Technology, University of Sciences, Hue University, 77 Nguyen Hue St., Hue, Vietnam
[2] HCMC University of Education, 280 An Duong Vuong, Ward 4, District 5, Ho Chi Minh City, Vietnam
[3] Ho Chi Minh City University of Food Industry, 140 Le Trong Tan Street, Tan Phu District, Ho Chi Minh City, Vietnam

** Liên hệ: thanhvt@hcmue.edu.vn*

Nhận bài: 20–06–2022; Ngày nhận đăng: 27–06–2022

Abstract. Multi-object image retrieval is a crucial problem in the field of image retrieval because of the diversity and complexity of digital images. In this paper, a method of multi-object image retrieval based on the R-CNN network with a KD-Tree structure is proposed to benefit from the advantages of the R-CNN network in identifying and classifying each object on the image separately; at the same time, the KD-Tree structure has high storage capacity and stable retrieval time. To solve this problem, we extracted the objects on the image data set, classified them with the R-CNN network model, and stored them on the KD-Tree structure. Then, each input image was segmented according to each object; the feature vector was extracted, and a similar image set was retrieved based on the KD-Tree structure. On this basis, a model of image retrieval using the R-CNN network and KD-Tree structure was proposed. To demonstrate the correctness of the proposed theoretical basis, we developed an experiment on the COCO image data set with an image retrieval precision of 0.6898. The experimental results were compared with other works on the same data set. This comparison proves the feasibility and effectiveness of the proposed method, which can be applied to multi-object images.

Keywords: R-CNN, KD-Tree, Multi-object image, image retrieval, similar images

1 Giới thiệu

Sự phát triển của các loại thiết bị điện tử làm cho dữ liệu đa phương tiện gia tăng nhanh theo thời gian, đặc biệt là ảnh đa đối tượng [1, 2]. Ngày nay, số lượng ảnh đa đối tượng gia tăng nhanh về số lượng và đa dạng về chủng loại thuộc nhiều lĩnh vực cũng là thách thức cho bài toán tìm kiếm ảnh đa đối tượng. Hơn nữa, việc xác định và bóc tách từng đối tượng riêng biệt trên ảnh đa đối tượng để có hiệu suất cao là một bài toán phức tạp. Sau khi phân đoạn và phân lớp từng đối tượng trên ảnh, việc lựa chọn một kỹ thuật học máy để thực hiện bài toán tìm kiếm ảnh để có hiệu suất truy vấn cao cũng là một thách thức. Vì vậy, bài toán tìm kiếm ảnh đa đối tượng được nhiều nhóm nghiên cứu quan tâm cải tiến và nâng cao hiệu suất và thời gian truy vấn ổn định. Hiện nay, có nhiều phương pháp để thực hiện quá trình phát hiện và phân loại từng đối tượng trên ảnh đa đối tượng như R-CNN (Region Convolutional Neural Network), Fast R-CNN, Faster R-CNN [12, 13]. Trong bài báo này, mỗi hình ảnh đầu vào được phân đoạn thành các vùng để nhận diện đối tượng bằng mô hình mạng R-CNN đồng thời phân loại từng đối tượng trên mỗi hình ảnh đã mang lại hiệu suất cao.

Hiệu suất tìm kiếm của bài toán truy vấn ảnh chịu ảnh hưởng của quá trình lưu trữ và tổ chức dữ liệu. Đồng thời, một cấu trúc dữ liệu lưu trữ hình ảnh là yếu tố ảnh hưởng đến thời gian tìm kiếm. Hiện nay, có một số cấu trúc dữ liệu dạng cây được ứng dụng nhiều trong bài toán tìm kiếm ảnh như S-Tree [3] và KD-Tree [4, 8, 14]. Trên cơ sở kế thừa cấu trúc dữ liệu đa chiều, KD-Tree được sử dụng cho quá trình lưu trữ để tìm kiếm tập ảnh tương tự với ảnh đầu vào được đánh giá là khả thi và hiệu quả thông qua các công trình [4, 9]. Trong bài báo này, mỗi hình ảnh sau khi thực hiện phân đoạn theo từng đối tượng, phân loại bằng mô hình R-CNN và trích xuất

vector đặc trưng được lưu trữ tại nút lá trên KD-Tree (*k-Dimensional Tree*). Tại mỗi nút lá là tập hợp các hình ảnh có độ tương tự gần nhau nhất. Cấu trúc KD-Tree được đánh giá với khả năng mở rộng số nút lá dễ dàng, phù hợp cho những bộ ảnh có khả năng mở rộng số phân lớp, mở rộng khả năng lưu trữ và thời gian tìm kiếm ổn định do cấu trúc KD-Tree đa nhánh cân bằng [9]. Vì vậy, để phát triển những ưu điểm từ việc phân lớp ảnh bằng mô hình mạng R-CNN và phương pháp lưu trữ, tìm kiếm ảnh trên cấu trúc KD-Tree nên một phương pháp kết hợp mạng R-CNN với cấu trúc KD-Tree được đề xuất thực hiện trong bài báo này là cần thiết và đúng đắn.

Đóng góp của bài báo gồm: (1) Trích xuất và phân lớp từng đối tượng trên ảnh bằng mạng R-CNN; (2) Trích xuất vector đặc trưng và xây dựng cấu trúc KD-Tree để lưu trữ dữ liệu hình ảnh đã phân đoạn; (3) Đề xuất mô hình tìm kiếm ảnh; xây dựng thực nghiệm trên bộ ảnh đa đối tượng COCO [5] và so sánh với một số công trình khác trên cùng bộ dữ liệu. Kết quả thực nghiệm cho thấy hiệu suất truy vấn ảnh dựa trên mô hình đề xuất là khá cao.

Phần còn lại của bài báo bao gồm: Phần 2 khảo các công trình nghiên cứu liên quan về trích xuất và phân loại đối tượng bằng mạng R-CNN, cấu trúc KD-Tree cho bài toán tìm kiếm ảnh; Phần 3 trình bày mô hình mạng R-CNN để phát hiện và phân lớp đối tượng; Phần 4 xây dựng cấu trúc KD-Tree để lưu trữ dữ liệu hình ảnh; Phần 5 đề xuất mô hình truy vấn ảnh; Phần 6 xây dựng thực nghiệm và đánh giá kết quả; kết luận và hướng phát triển tiếp theo được trình bày trong Phần 7.

2 Các công trình liên quan

Trong bài báo này, quá trình tìm kiếm ảnh đa đối tượng được thực hiện qua các giai đoạn gồm: (1) Trích xuất các đối tượng thị giác trên ảnh và phân lớp đối tượng; (2) xây dựng cấu trúc KD-Tree lưu trữ và tìm kiếm ảnh tương tự dựa trên cấu trúc dữ liệu đã xây dựng. Vì vậy, một số công trình được khảo sát về trích xuất và phân loại đối tượng mạng R-CNN và tìm kiếm ảnh bằng cấu trúc KD-Tree nhằm phân tích ưu nhược điểm của từng phương pháp để đưa ra phương pháp kết hợp mạng R-CNN và cấu trúc KD-Tree để giải bài toán tìm kiếm ảnh đa đối tượng và nâng cao hiệu suất truy vấn.

Chiao và cs. [6] đã thực hiện một phương pháp phát hiện và phân loại các khối u vú sử dụng mặt nạ R-CNN trên ảnh siêu âm. Mục đích của bài báo là xây dựng mô hình phát hiện, phân đoạn và phân loại tự động các tổn thương vú bằng hình ảnh siêu âm. Dựa trên kỹ thuật học sâu, một kỹ thuật sử dụng các vùng mặt nạ với mạng lưới thần kinh phức hợp đã được phát triển để phát hiện tổn thương và phân biệt giữa lành tính và ác tính trên hình ảnh. Độ chính xác trung bình trung bình là 0,75 cho việc phát hiện và phân đoạn. Độ chính xác tổng thể của phân loại lành tính và ác tính trên hình ảnh là 0,85. Công trình này được đánh giá là khả thi và ứng dụng tốt cho

lĩnh vực phát hiện sớm bệnh ung thư vú qua hình ảnh bằng mạng R-CNN. Bên cạnh đó, Kuznetsova và cs. [7] đã đề xuất một phương pháp phân tích ngữ nghĩa trực quan hình ảnh. Công trình này đã trình bày một bộ sưu tập gồm 9,2 triệu hình ảnh (COCO, PASCAL VOC) được chú thích thống nhất để phân loại hình ảnh và phát hiện đối tượng bằng mô hình mạng R-CNN. Sau đó, các mối quan hệ trực quan giữa các đối tượng được xác định dựa trên trên ảnh đầu vào. Phương pháp đề xuất này được đánh giá là khả thi, hiệu quả và áp dụng cho nhiều bộ ảnh đa đối tượng khác nhau.

Ram và cs. [10] sử dụng kỹ thuật tìm kiếm láng giềng k-NN dựa trên cấu trúc KD-Tree. Việc kết hợp này nhằm cải tiến hiệu suất tìm kiếm bằng cách xây dựng cây phân vùng không gian ngẫu nhiên để thực hiện các lược đồ tìm kiếm theo cấu trúc KD-Tree. Tác giả đã chứng minh tính hiệu quả về thời gian truy vấn cũng như hiệu suất tìm kiếm. Trong công trình này, tác giả đề cập tới hai cải tiến: (1) cải thiện độ phức tạp tổng thể giải thuật tìm kiếm; (2) thực hiện đa chỉ mục trên cây KD-Tree để nâng cao hiệu quả tìm kiếm về mặt thời gian. Cùng thời điểm này, Chen và cs. [11] đã sử dụng hai kỹ thuật tìm kiếm láng giềng RNN (*Range Nearest Neighbors*) và NN (*Nearest Neighbors*) dựa trên cây KD-Tree. Kỹ thuật RNN nhằm giảm các tính toán khoảng cách không cần thiết bằng cách kiểm tra vị trí của đối tượng đang xét nằm bên trong hay bên ngoài vùng lân cận của điểm cần tìm. Kỹ thuật NN được sử dụng để giảm các nút truy cập dư thừa bằng cách lưu chỉ số truy cập các điểm láng giềng. Thực nghiệm chứng minh tính hiệu quả của việc kết hợp các thuật toán tìm kiếm láng giềng RNN, NN và kNN trên cây KD-Tree là hiệu quả.

Zhang [17] và cs. đã thực hiện xây dựng cấu trúc Vocabulary-KDTree nhằm thực hiện bài toán đối sánh hình ảnh. Trong công trình này, nhóm tác giả đã thực hiện hai quá trình: (1) phân cụm dữ liệu hình ảnh theo tính chất tương đồng và (2) đối sánh dữ liệu trực tuyến với một ảnh đầu vào. Cấu trúc Vocabulary-KDTree dựa trên đặc trưng SIFT (*Scale-Invariant Feature Transform*) bằng cách điều chỉnh trọng số tại các nút trên cây. Cấu trúc Vocabulary-KDTree được chia thành hai nhóm: (1) nhóm chứa các đặc trưng hình ảnh và (2) nhóm các nút lá thực hiện điều chỉnh các trọng số liên quan đến quá trình huấn luyện để xây dựng Vocabulary-KDTree. Mô hình truy vấn ảnh được thực hiện theo hai pha. Tại pha offline, mỗi hình ảnh sau khi trích xuất đặc trưng được đối sánh và gom cụm với cấu trúc KD-Tree; từ đó xây dựng cây Vocabulary KD-Tree và thực hiện gom cụm lại trên cấu trúc này. Tại pha online, một ảnh đầu vào sau khi trích xuất đặc trưng được so sánh đặc trưng này với cấu trúc Vocabulary KD-Tree, tìm ra từ khóa làm cơ sở so sánh với đặc trưng đã trích xuất. Cuối cùng, lọc bỏ những bất thường trong kết quả tìm kiếm và trả về kết quả tốt nhất.

Narasimhulu và cs. [18] đã đề xuất một phương pháp tìm kiếm ảnh tượng tự dựa trên cấu trúc KD-Tree. Từ một ảnh đầu vào thực hiện tìm kiếm trên cấu trúc KD-Tree bằng thuật toán tìm kiếm theo số láng giềng nhiều nhất để làm căn cứ xác định phân lớp cho hình ảnh. Cuối cùng, tác

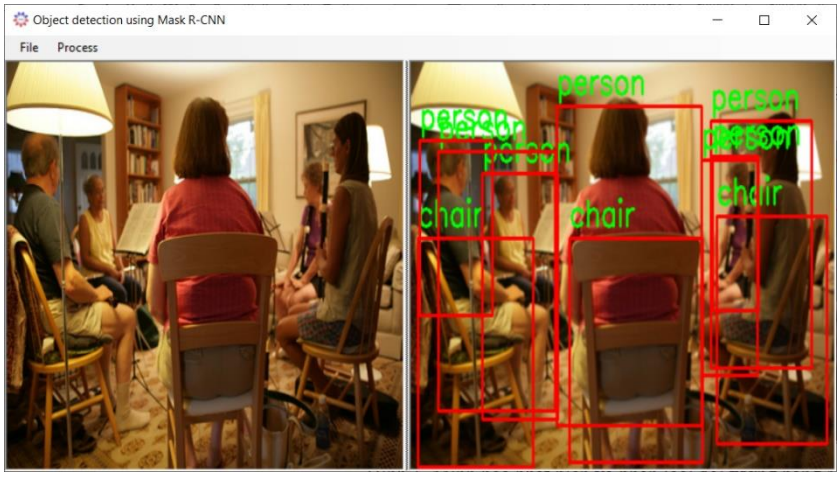
giả dùng thang đo khoảng cách để thực hiện phân lớp các tập dữ liệu hình ảnh huấn luyện. Trong công trình này, cây KD-Tree được sử dụng trực tiếp để lưu trữ dữ liệu và phân lớp cho một ảnh đầu vào với kết quả tốt mà không mất nhiều chi phí trung gian. Đây là một mô hình được đề xuất cho bài toán phân lớp và tìm kiếm ảnh dựa vào cấu trúc KD-Tree và được đánh giá là khá tốt.

Những công trình nghiên cứu trên cho thấy tính khả thi cho bài toán trích xuất và phân loại đối tượng bằng R-CNN và tìm kiếm ảnh bằng cấu trúc KD-Tree. Tuy nhiên, sự kết hợp giữa kỹ thuật R-CNN và cấu trúc KD-Tree để nâng cao hiệu quả cho bài toán tìm kiếm ảnh đa đối tượng còn hạn chế về số lượng. Vì vậy, trong bài báo này, một mô hình trích xuất, phân loại đối tượng, trích xuất vector đặc trưng và lưu trữ trên cấu trúc KD-Tree được áp dụng cho bài toán tìm kiếm ảnh đa đối tượng và được thực hiện nhằm kết hợp những ưu điểm hiện có của kỹ thuật mạng R-CNN và cấu trúc KD-Tree.

3 Mạng R-CNN phát hiện và phân lớp đối tượng

Nâng cao hiệu suất phát hiện đối tượng là một trong những nhiệm vụ thách thức trong thị giác máy tính. Hiện nay có nhiều công trình sử dụng mạng R-CNN [1], Fast R-CNN và Faster R-CNN [12] để phát hiện các đối tượng riêng biệt trên ảnh. Mục đích của quá trình phát hiện đối tượng là phân loại đối tượng, nhận dạng đối tượng, nhận dạng mẫu, định vị đối tượng trên ảnh, tìm mối quan hệ giữa các đối tượng trên ảnh, v.v. Vì vậy, mạng R-CNN là một kỹ thuật tiên tiến, được sử dụng rộng rãi trong các công trình đã công bố trong những năm gần đây. Kiến trúc của mạng R-CNN gồm ba thành phần: (1) trích xuất vùng đề xuất đối tượng (*Region proposal*), có tác dụng tạo và trích xuất các vùng chứa vật thể được bao bởi các bounding box; (2) trích xuất đặc trưng (*Feature Extractor*), giúp nhận diện hình ảnh từ các region proposal thông qua mạng CNN; (3) phân loại (*classifier*), dựa trên ảnh đầu vào là các đặc trưng để phân loại hình ảnh chứa trong vùng đề xuất về đúng nhãn (Hình 1) [1, 12, 13].

Mạng R-CNN được đánh giá là hiệu quả cho các bài toán phát hiện đối tượng, phân loại đối tượng trên ảnh do những ưu điểm như hiệu suất phát hiện đối tượng và phân loại đối tượng cao; một ưu điểm khác là mạng R-CNN có thể trích xuất các tính năng của hình ảnh một cách tự động. Tuy nhiên, nhược điểm của phương pháp phát hiện và phân loại đối tượng trên R-CNN là nó phải vượt qua nhiều giai đoạn độc lập trong đó có trích xuất đặc trưng từ một mạng CNN trên từng vùng đề xuất tạo từ vùng chứa ảnh. Trong bài báo này, ưu điểm của mạng R-CNN được ứng dụng để phát hiện và phân loại từng đối tượng trên ảnh với hiệu suất phân loại cao. Quá trình phát hiện và phân loại đối tượng trên ảnh bằng mạng Mask R-CNN được minh họa trên Hình 1.



Hình 1. Minh họa phát hiện và phân loại đối tượng bằng Mask R-CNN

Mỗi hình ảnh sau khi trích xuất từng đối tượng trên ảnh bằng mạng R-CNN là kết quả của quá trình trích xuất vector đặc trưng của ảnh phân đoạn. Trên cơ sở này, mỗi vùng ảnh được trích xuất đặc trưng theo các nhóm đặc trưng về diện tích, chu vi, màu sắc, hình dạng và kết cấu gồm 81 thành phần cho mỗi vùng ảnh. Quá trình trích xuất vector đặc trưng có 81 thành phần được kế thừa từ công trình [4] và minh họa trên Hình 2.

Hình 2. Minh họa trích xuất vector đặc trưng hình ảnh gồm 81 thành phần

4 Cấu trúc KD-Tree cho tìm kiếm ảnh đa đối tượng

4.1 Xây dựng cấu trúc KD-Tree lưu trữ ảnh phân đoạn

Mỗi hình ảnh sau khi phân đoạn đối tượng và trích xuất thành các vector đặc trưng được lưu trữ trên cấu trúc KD-Tree [14]. Mục đích xây dựng cấu trúc KD-Tree để lưu trữ hình ảnh đã phân đoạn là làm cho quá trình tìm kiếm nhanh và hiệu quả. Bên cạnh đó, cấu trúc KD-Tree có khả năng mở rộng số nhánh đã được chứng minh từ công trình [9]. Vì vậy, trong bài báo này, một cải tiến khác là mỗi hình ảnh được phân đoạn theo đối tượng bằng mạng R-CNN trước khi lưu trữ trên cấu trúc KD-Tree. Theo đó, nghiên cứu này đề xuất thuật toán xây dựng cấu trúc KD-Tree dựa trên tập vector đặc trưng vùng ảnh đối tượng được đề xuất. Trong thuật toán 1, hàm ExtractFeature được kế thừa từ công trình [4] còn hàm RCNN được thực hiện để phân đoạn ảnh dựa trên mạng R-CNN.

Thuật toán 1: Xây dựng cấu trúc KD-Tree

Input: Image data set COCO

Output: KD-Tree

Function **BKDT** (F, W, h, n)

Begin

Initialize height h, number of branches n;

W = Initialize(random a set of vectors weight);

Nodei.w = W_i;

Segment-image = **RCNN**(Image I);

F_i = **ExtractFeature**(Segment-image);

KD-Tree = **Initialze**(F_i, W, h, n);

Insert each vector F_i into KD-Tree;

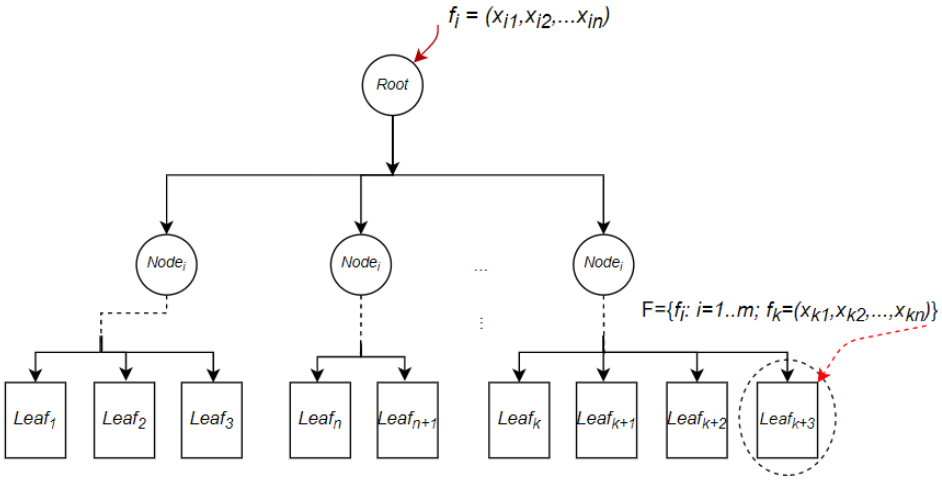
Return KD-Tree;

End.

Gọi n là số phần tử trong tập F để thực hiện xây dựng cây KD-Tree và h là chiều cao của cây. Khi xây dựng cây KD-Tree, thuật toán 1 cho phép thêm n phần tử vào cây có chiều cao là h. Cây KD-Tree là cây cân bằng nên khi thêm phần tử vào cây, mọi phần tử đều phải được duyệt từ

nút gốc đến nút lá. Vì vậy, chi phí để xây dựng cây KD-Tree chiều cao h có n phần tử là $O(n \times h)$. Vì h là hằng số, nên độ phức tạp của thuật toán 1 là $O(n)$.

Sau khi xây dựng cấu trúc KD-Tree gồm một nút gốc (*Root*) và các nút trong ($Node_i$) lưu trữ tập vector trọng số, các nút lá (*Leaf*) lưu trữ tập vector hình ảnh có độ tương tự gần nhau nhất. Cấu trúc KD-Tree được minh họa trên Hình 3.



Hình 3. Minh họa cấu trúc KD-Tree

4.2 Huấn luyện cấu trúc KD-Tree

Ban đầu cấu trúc KD-Tree được xây dựng với bộ vector trọng số lưu trữ tại các nút trong là ngẫu nhiên nên hiệu suất phân bổ ảnh tương tự tại nút lá chưa cao. Vì vậy, cần phải điều chỉnh vector tại các nút trong của KD-Tree để quá trình chèn vector đặc trưng hình ảnh vào KD-Tree sao cho nút lá chứa các hình ảnh cùng một phân lớp là nhiều nhất. Thuật toán 2 kế thừa các hàm SetLabel2Leaf và UpdateWeight từ công trình [4, 9]. Quá trình huấn luyện vector trọng số được thực hiện liên tiếp theo các tập $Epoch_i$ dữ liệu và được trình bày như sau:

Thuật toán 2: Huấn luyện trọng số trên KD-Tree

Input: Set of initialized weights, $Epoch_i$
Output: Set of training weights
Function TKDT (*InitW eight*, $Epoch_i$)
Begin

Weight = *InitWeight*();
repeat

BKDT(*Epoch$_i$, InitW eight, h, n*);

SetLabel2Leaf (*KD-Tree, ListLabels*);
Pi = SumofV ectorRightLabel()/*SumofV ectorinEpoch$_i$* ();
Road (*f.wrong*) = *LeafWrong.Road;*
Road (*f.right*) = *LeafRight.Road;*
Get(*Nodew*)*inRoadf,w*;
NewW eight = **UpdateWeight**(*Nodew, Roadf,w*);
BKDT (*Epoch$_i$, NewW eight, h, n*);
SetLabel2Leaf (*KD-Tree, ListLabels*);
Pj = SumofV ectorRightLabel()/*SumofV ectorinEboli*();

until (*Pj* < *Pi*);

Weight = NewWeight;

foreach (*SubTree*) **do**

F = Find(*fj* **in** *SubTree*);

WeightSub = **TKDT** (*InitWeight, SubTree.F*);
endForeach
Weight = *NewWeight* ∪ *WeightSub*;
Return *Weight*;

End.

Gọi *p* là số lần điều chỉnh véc-tơ trọng số; *h* là chiều cao cây; *m* là số phần tử tham gia vào quá trình xây dựng cây theo từng *Epoch$_i$*. Quá trình huấn luyện cấu trúc KD-Tree được thực hiện thông qua việc cập nhật trọng số để tạo cây và gán nhãn tại nút lá. Vì vậy, chi phí để thực hiện thực hiện thuật toán 2 là $(p \times h \times m)$. Vì *p*, *h* là các hằng số nhỏ nên độ phức tạp của thực hiện thuật toán 2 là O(*m*)**.**

5 Mô hình tìm kiếm ảnh dựa trên R-CNN và KD-Tree

5.1 Mô hình đề xuất

Trên cơ sở kết hợp mạng R-CNN và cấu trúc KD-Tree để áp dụng cho bài toán tìm kiếm ảnh đa đối tượng, chúng tôi đề xuất mô hình tìm kiếm ảnh (Hình 4).

Mô hình tìm kiếm ảnh đa đối tượng dựa trên mạng R-CNN và cấu trúc KD-Tree gồm hai pha: Pha tiền xửa lý và pha truy vấn với các bước như sau:

(1) Phát hiện và phân loại đối tượng trên ảnh bằng mạng R-CNN

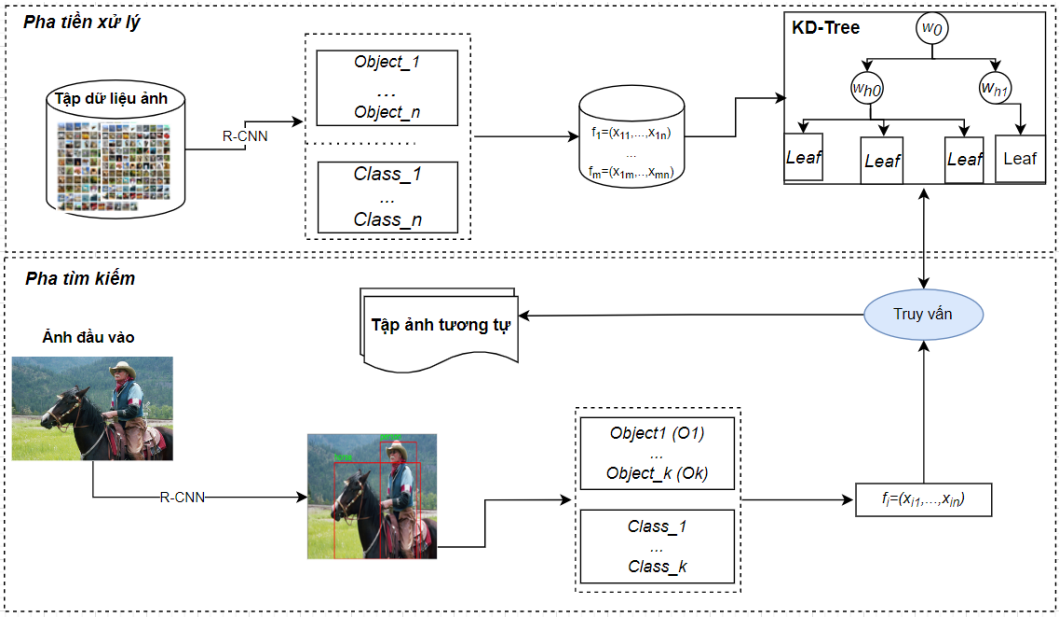
(2) Trích xuất vector đặc trưng hình ảnh đã phân đoạn

(3) Xây dựng cấu trúc KD-Tree lưu trữ hình ảnh

(4) Ảnh đầu vào được phát hiện và phân loại bằng mạng R-CNN

(5) Trích xuất vector đặc trưng cho ảnh đầu vào theo từng vùng đối tượng

(6) Tìm kiếm trên KD-Tree để trích xuất tập ảnh tương tự với ảnh đầu vào



Hình 4. Mô hình tìm kiếm ảnh được đề xuất trúc KD-Tree

5.2 Thuật toán tìm kiếm ảnh tương tự dựa trên cấu trúc KD-Tree

Sau khi xây dựng cấu trúc KD-Tree, các nút lá lưu trữ tập dữ liệu hình ảnh. Vì vậy, quá trình tìm kiếm tập ảnh tương tự với một ảnh đầu vào (I) cần phải duyệt từ nút gốc đến nút lá. Nếu các vector đặc trưng của vùng ảnh phân đoạn của ảnh I thuộc về một nút lá $leaf_k$ thì trích xuất tập ảnh tương tự là tập ảnh tại nút lá $leaf_k$. Trong trường hợp ảnh I có nhiều ảnh phân đoạn I_1, …, I_n và các vector đặc trưng của ảnh I thuộc nhiều nút lá khác nhau thì tập ảnh tương tự với ảnh I chính là tập các ảnh thuộc tập các nút lá mà f_{ik} tìm được.

Thuật toán 3: Tìm kiếm ảnh tương tự dựa trên KD-Tree và mạng R-CNN

Input: Tập vector đặc trưng F= { f_{Ii} } của ảnh I, KD-Tree

Output: Tập ảnh tương tự CI

Function RKDT(F, $KD-Tree$)
Begin
$CI = \varnothing$;
Foreach ($f_i \in$ F) **do**
Browsing from root to leaf on KD-Tree;
If ($f_i \in leaf_k$) **then**
CI = $leaf_k$.{f_k};
Endif;
EndForeach;
Return CI;
End.

Gọi h là chiều cao của cấu trúc KD-Tree; k là số nhánh tối đa tại $Node_i$ bất kỳ, dữ liệu đầu vào là vector đặc trưng f_i có n chiều. Khi truyền vector f_i vào KD-Tree, thuật toán 3 duyệt qua các mức của cây. Tại mỗi mức trên KD-Tree chọn một nút tốt nhất và đi theo hướng đã chọn. Do đó, tại mỗi mức có tối đa k phép so sánh để chọn nút tốt nhất. Mỗi lần so sánh thuật toán 3 duyệt qua n phần tử của vector f_i. Vì vậy, tại mỗi mức số phép toán tối đa là $k \times n$. Cây có chiều cao h, nên số phép toán tối đa để duyệt từ gốc đến lá theo một hướng được chọn là $k \times n \times h$. Vì h, k là hằng số nhỏ, nên độ phức tạp của thuật toán phụ thuộc vào n. Mặc khác, số chiều vector f_i là cố định ban đầu nên n cũng là hằng số. Gọi C là giá trị hằng số và $k \times h \times n < C$ nên $k \times h \times n \leq C \times 1$. Vậy độ phức tạp của thuật toán 3 là O(1).

6 Thực nghiệm và đánh giá

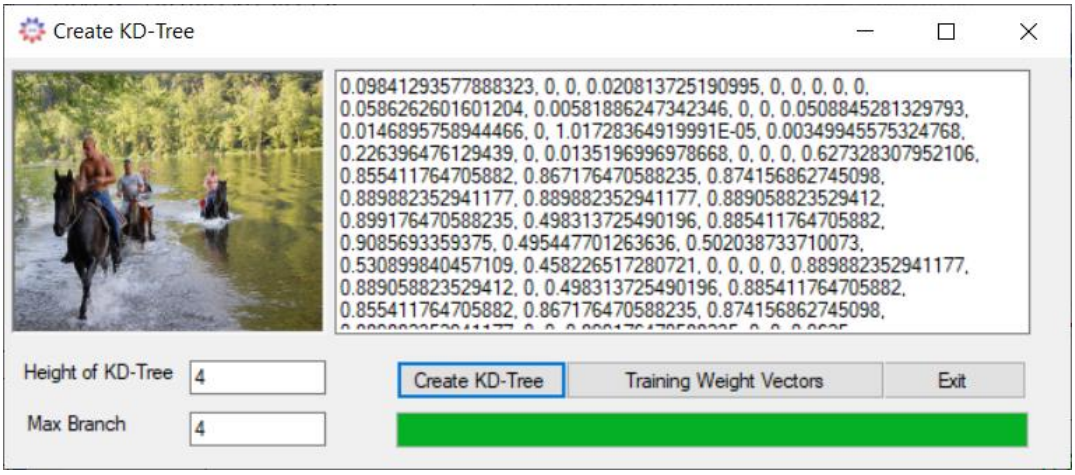
6.1 Dữ liệu và môi trường thực nghiệm

Bộ ảnh COCO (*Microsoft Common Objects in Context*) là bộ ảnh đa đối tượng, đa dạng và có nhiều phân lớp và được sử dụng để phát hiện đối tượng, phân đoạn, phát hiện điểm chính và phụ đề quy mô lớn. Bộ dữ liệu bao gồm 163.957 hình ảnh. Sau khi thực hiện phân lớp ảnh bằng mạng R-CNN, bộ ảnh COCO có 79 phân lớp được sử dụng cho thực nghiệm trên tập Validation gồm 5.000 ảnh.

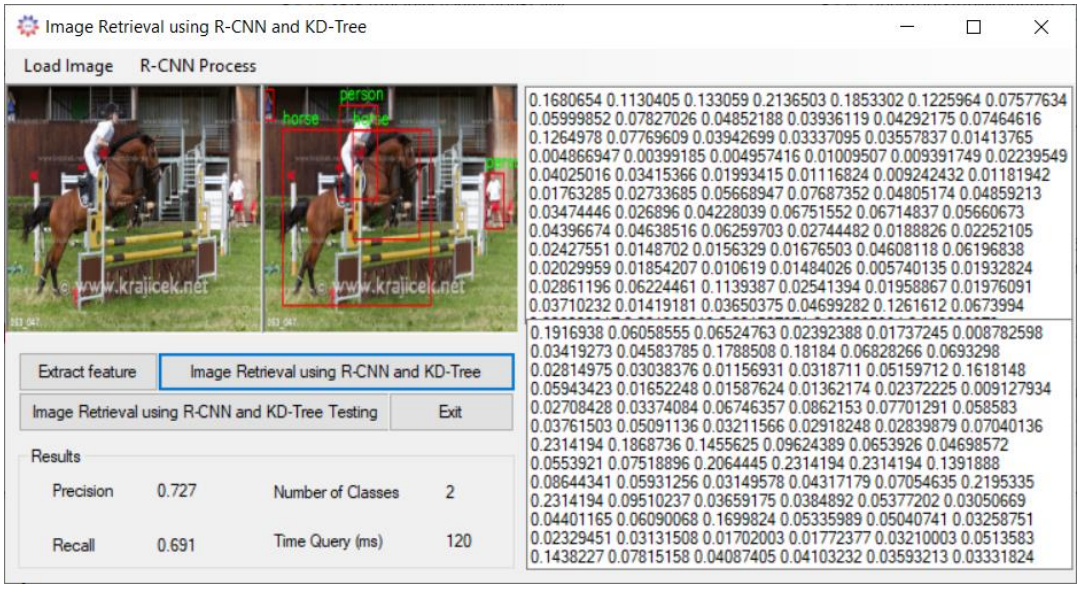
Môi trường thực nghiệm hệ tìm kiếm ảnh tương tự (IR-KDT**)** được xây dựng trên nền tảng dotNET Framework 4.5, ngôn ngữ lập trình C#. Cấu hình máy tính: Intel(R) Core™ i5-5200U, CPU 2.7GHz, RAM 16GB và hệ điều hành Windows 10 Professional.

6.2 Xây dựng thực nghiệm

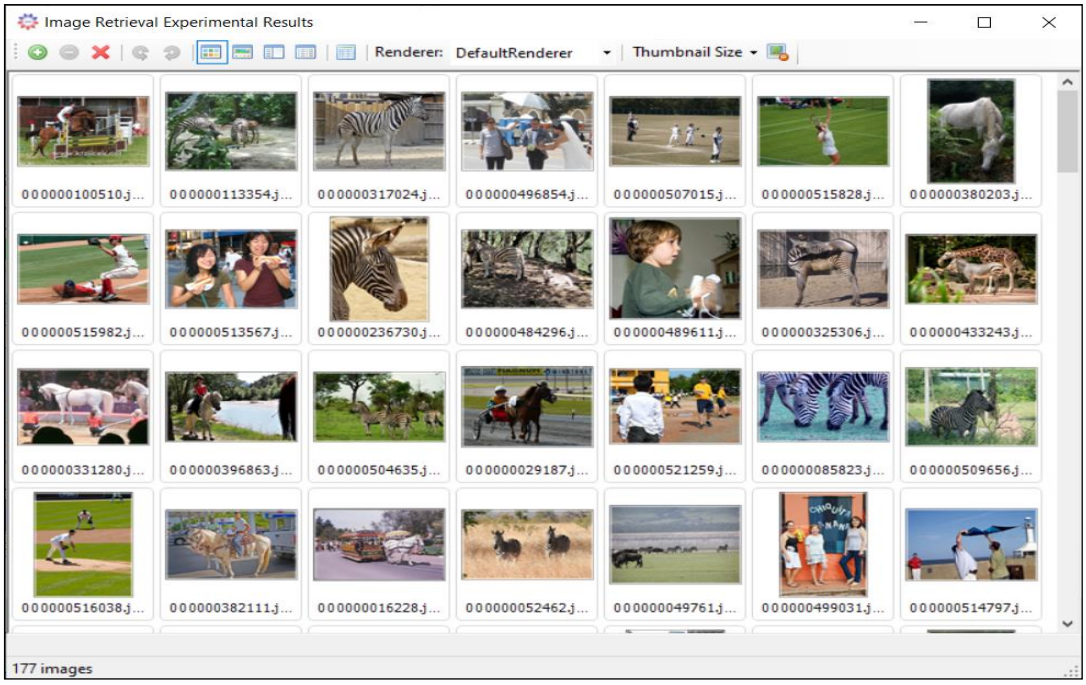
Thực nghiệm xây dựng hệ truy vấn ảnh IR-KDT gồm xây dựng cấu trúc KD-Tree để lưu trữ hình ảnh của bộ COCO đã phân đoạn tại các nút lá (Hình 5). Dựa trên số phân lớp của bộ ảnh COCO để xác định số nhánh và chiều cao phù hợp cho KD-Tree. Sau khi xây dựng xong cấu trúc KD-Tree, chúng tôi tìm tập ảnh tương tự dựa trên KD-Tree và R-CNN (Hình 6). Một tập ảnh tương tự với một ảnh đầu vào 000000100510.jpg (bộ ảnh COCO) được minh họa trên Hình 7.



Hình 5. Xây dựng KD-Tree



Hình 6. Hệ tìm kiếm ảnh IR-KDT



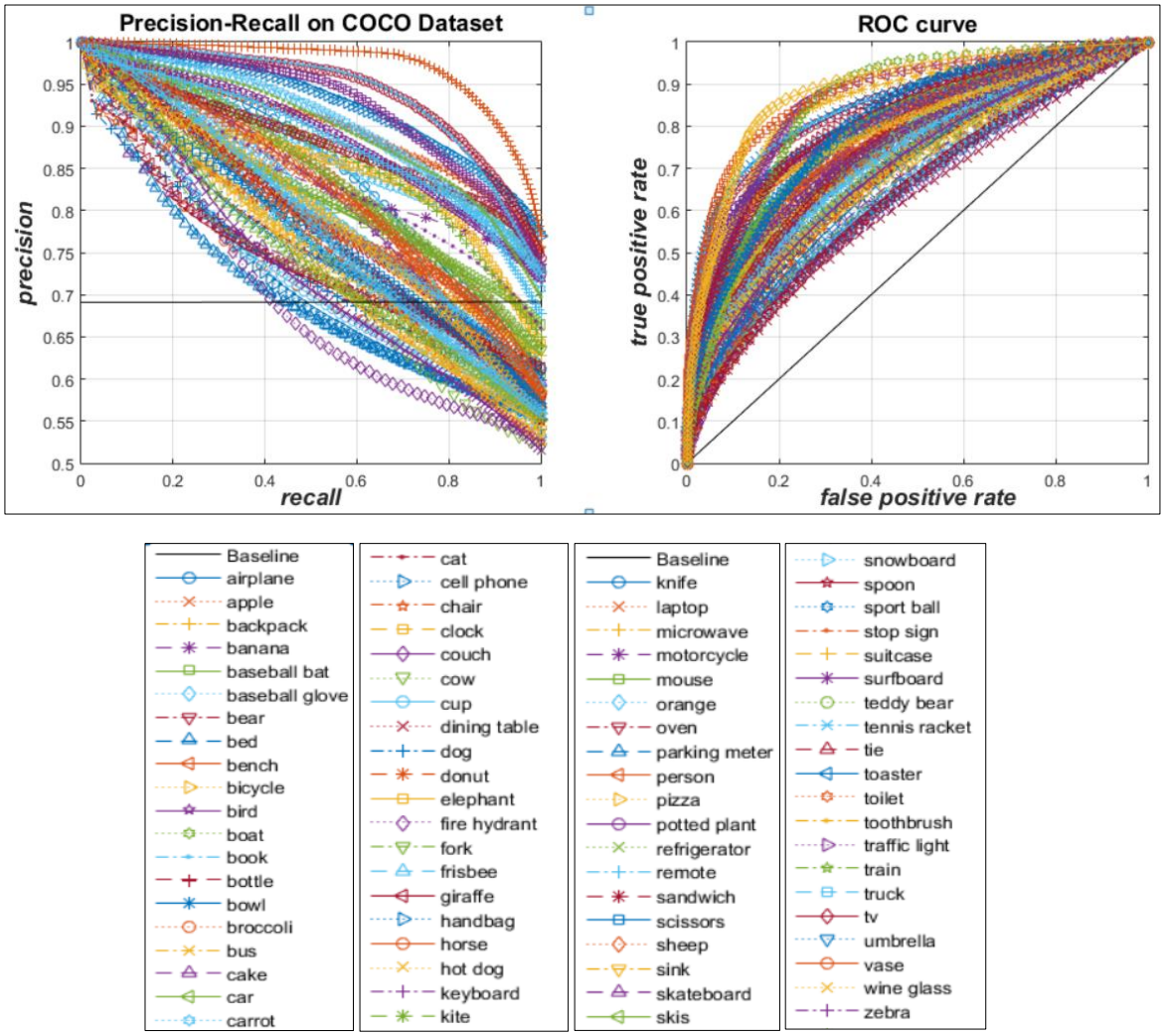
Hình 7. Kết quả truy vấn trên KD-Tree sử dụng R-CNN

Kết quả thực nghiệm hệ tìm kiếm ảnh IR-KDT bao gồm độ chính xác trung bình (*Precision*), độ phủ (*Recall*), độ dung hòa (*F-measure*) và thời gian truy vấn trung bình (*Time query*) tính bằng mili giâycủa hệ truy vấn ảnh IR-KDT (Bảng 1).

Bảng 1. Hiệu suất tìm kiếm ảnh của hệ IR-KDT

Tập ảnh	Precision	Recall	F-measure	Time query (ms)
COCO	0,6898	0,6472	0,6678	109,02

Kết quả thực nghiệm đánh giá mô hình truy vấn ảnh đề xuất được thực hiện trên 5.000 ảnh với bộ dữ liệu COCO. Độ chính xác trung bình được lấy theo TopK ($K = 5$). Kết quả tìm kiếm ảnh trung bình trong hệ IR-KDT được minh họa bằng đồ thị đường cong ROC trên Hình 8. Mỗi đường cong trên đồ thị mô tả kết quả truy vấn với độ chính xác và độ phủ của một chủ đề ảnh trong bộ dữ liệu COCO. Đồng thời, đường cong tương ứng trong đồ thị ROC cho biết tỷ lệ kết quả truy vấn đúng và sai, nghĩa là diện tích dưới đường cong này đánh giá được tính đúng đắn của các kết quả truy vấn. Đồ thị cho thấy tính chính xác của hệ truy vấn tập ảnh COCO nằm tập trung ở vùng [0,52, 1,0]. Đồ thị đường cong ROC biểu diễn các giá trị true positive và false positive theo độ phủ recall. Các giá trị nằm tập trung trên đường cơ sở; nhiều giá trị nằm trong vùng true positive hơn vùng false positive.



Hình 8. Đường cong Precision, Recall và ROC cho bộ ảnh COCO

6.3 Đánh giá kết quả thực nghiệm

Hiệu suất của hệ tìm kiếm ảnh dựa trên mối quan hệ ngữ nghĩa (IR-KDT) được so sánh với các công trình khác trên cùng bộ ảnh COCO. Kết quả so sánh được trình bày trong Bảng 2.

Kết quả tìm kiếm ảnh tương tự của hệ IR-KDT cao hơn các công trình khác trên cùng bộ dữ liệu. Điều này cho thấy phương pháp đề xuất của chúng tôi là khả thi, hiệu quả và có thể so sánh với các công trình khác cùng lĩnh vực bởi vì các lý do sau: (1) hệ IR-KDT sử dụng kỹ thuật mạng R-CNN để phân đoạn ảnh và phân lớp đối tượng nên hiệu suất phân lớp cao làm tiền đề cho quá trình tìm kiếm trên KD-Tree; (2) quá trình huấn luyện tập vector trọng số trên KD-Tree

để lưu trữ tập ảnh tương tự tại nút lá giúp quá trình hội tụ các ảnh tương tự tại nút lá là tốt nhất. Bên cạnh đó, sự kết hợp giữa kỹ thuật R-CNN và cấu trúc KD-Tree giúp giải quyết bài toán tìm kiếm ảnh đa đối tượng hiệu quả và thời gian tìm kiếm ổn định.

Bảng 2. So sánh hiệu suất truy vấn ảnh hệ IR-KDT với các công trình khác trên bộ ảnh COCO

Phương pháp thực hiện	Độ chính xác trung bình
CN MAX, TopK = 5, [15]	0,391
CAMP, TopK = 5, [16]	0,689
IR-KDT	**0,6898**

7 Kết luận và hướng phát triển

Trong bài báo này, hệ truy vấn ảnh đa đối tượng IR-KDT được thực hiện dựa trên cấu trúc KD-Tree và mạng R-CNN. Một số kết quả đạt được gồm: (1) Thực hiện trích xuất và phân loại từng đối tượng trên ảnh bằng mạng R-CNN; (2) xây dựng cấu trúc KD-Tree để lưu trữ tập ảnh phân đoạn sau khi trích xuất đặc trưng; (3) đề xuất mô hình truy vấn ảnh dựa trên mạng R-CNN và cấu trúc KD-Tree đã xây dựng; (4) đề xuất các thuật toán xây dựng KD-Tree, huấn luyện trọng số, tìm kiếm trên KD-Tree; (4) đề xuất mô hình tìm kiếm ảnh tương tự; (5) thực nghiệm trên bộ ảnh COCO với độ chính xác tìm kiếm ảnh trung bình là 0,6898 và so sánh với các công trình khác trên cùng bộ ảnh. Trong hướng phát triển tiếp theo, chúng tôi sẽ kết hợp mạng R-CNN trích xuất và phân loại đối tượng, sau đó xây dựng mối quan hệ giữa các đối tượng trên ảnh bằng cấu trúc KD-Tree và thực hiện tìm kiếm ảnh theo ngữ nghĩa dựa trên ontology nhằm nâng cao hiệu suất cho bài toán truy vấn ảnh.

Lời cảm ơn

Nhóm tác giả trân trọng cảm ơn Khoa Công nghệ thông tin, Trường Đại học Khoa học, Đại học Huế, nhóm nghiên cứu SBIR-HCM, Trường Đại học Sư phạm Tp. HCM và Trường Đại học Công nghiệp Thực phẩm Thành phố Hồ Chí Minh đã hỗ trợ về chuyên môn và cơ sở vật chất để nhóm tác giả hoàn thành nghiên cứu này. Bài báo là kết quả của đề tài thực hiện bằng nguồn kinh phí hỗ trợ từ Chương trình Vườn ươm Sáng tạo Khoa học và Công nghệ trẻ, Thành Đoàn thành phố Hồ Chí Minh và Sở Khoa học và Công nghệ thành phố Hồ Chí Minh, theo hợp đồng số "33/2021/HĐ-KHCNT-VƯ" ký ngày 8 tháng 12 năm 2021.

Tài liệu tham khảo

1. Chen, S., Li, Z., and Tang, Z., (2020), Relation r-cnn: A graph based relation-aware network for object detection, *IEEE Signal Processing Letters*, *27*, 1680-1684.
2. He, K., Gkioxari, G., Dollár, P., and Girshick, R., (2017), Mask r-cnn, In *Proceedings of the IEEE international conference on computer vision*, 2961-2969.
3. Le, T. M., and Van, T. T., (2015), Image Retrieval System Base on EMD Similarity Measure and S-Tree, *arXiv preprint arXiv: 1506.01165*.
4. Nguyv preprint arThu Thành Văn, Mt arXiv: 1506.01165 B Phân l Văn, Mt arXiv: 1506.01165 Base on EMD Similarity Measure an*Các công trình nghiên c506.01165 Base on EMD Similarity Measure and S-Tree, tection, ,* 40-52.
5. https://cocodataset.org/#download, 30/5/2022.
6. Chiao, J. Y., Chen, K. Y., Liao, K. Y. K., Hsieh, P. H., Zhang, G., and Huang, T. C., (2019), Detection and classification the breast tumors using mask R-CNN on sonograms, *Medicine*, *98*(19).
7. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., and Ferrari, V., (2020), The open images dataset v4, *International Journal of Computer Vision*, *128*(7), 1956-1981.
8. Zhang, Y., Wang, N., Zhang, S., Li, J., and Gao, X., (2016), Fast face sketch synthesis via kd-tree search, In *European Conference on Computer Vision*, Springer, Cham, 64-77.
9. Nguy7.nger, Cham,Thu Thành Văn, MCham, e on Computer Visiona kd-tree search, In I., Pont-Tuset, J., and Ferrari, Vnghệ Trẻ, Thành Đoàn thành ph*Khành Văn, MCham, e on Computer Visiona kd-tree search, In I., FAIR21), ĐH Công nghip://fair.conf.vn/" \t "_blank" e search, In I., Pont-Tuset, J., and Ferrari, Vnghệ Trẻ, T* DOI: 10.15625/vap.2021.0075
10. Ram, P., & Sinha, K., (2019), Revisiting kd-tree for nearest neighbor search, In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining*,1378-1388.
11. Chen, Y., Zhou, L., Tang, Y., Singh, J. P., Bouguila, N., Wang, C., and Du, J., (2019), Fast neighbor search by using revised kd tree, *Information Sciences*, *472*, 145-162.
12. Lee, H., Eum, S., and Kwon, H., (2019), Me r-cnn: Multi-expert r-cnn for object detection, *IEEE Transactions on Image Processing*, *29*, 1030-1044.
13. Schroeder, B., & Tripathi, S., (2020), Structured query-based image retrieval using scene graphs, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 178-179.
14. Bentley, J. L., (1975), Multidimensional binary search trees used for associative searching, *Communications of the ACM*, *18*(9), 509-517.
15. Icarte, R. T., Baier, J. A., Ruz, C., and Soto, A., (2017), How a general-purpose commonsense ontology can improve performance of learning-based image retrieval, *arXiv preprint arXiv:1705.08844*.
16. Wang, Z., Liu, X., Li, H., Sheng, L., Yan, J., Wang, X., and Shao, J., (2019), Camp: Cross-modal adaptive message passing for text-image retrieval, In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5764-5773.

17. Zhang, F., Gao, Y., & Xu, L., (2020), An adaptive image feature matching method using mixed Vocabulary-KD tree, *Multimedia Tools and Applications*, *79*(23), 16421-16439.

18. Narasimhulu, Y., Suthar, A., Pasunuri, R., and Venkaiah, V. C., (2021), CKD-Tree: An Improved KD-Tree Construction Algorithm, In *ISIC*, 211-218.