

ĐÁNH GIÁ HIỆU SUẤT TRUY VẤN ẢNH TRÊN CÁC BIẾN THỂ KD-TREE

Nguyễn Thị Định^{1,3*}, Lê Mạnh Thạnh¹, Văn Thế Thành²

¹Khoa Công nghệ thông tin, Trường Đại học Khoa học, Đại học Huế

²Trường Đại học Sư phạm TP. HCM

³Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Thực phẩm TP. HCM

*Email: dinhnt@hufi.edu.vn

Ngày nhận bài: 4/4/2022; ngày hoàn thành phản biện: 12/5/2022; ngày duyệt đăng: 22/6/2022

TÓM TẮT

Trong bài báo này, một khảo sát về đánh giá hiệu suất truy vấn ảnh trên một số biến thể KD-Tree được trình bày. Có nhiều biến thể KD-Tree được sử dụng cho các bài toán về xử lý dữ liệu không gian đa chiều trong đó có dữ liệu ảnh số như Randomly Projected K-d Trees, Buffer k-d Trees, Progressive k-d Tree v.v. Trong đó, k-NN KD-Tree và Distributed KD-Tree là hai biến thể được khảo sát về phương pháp xây dựng, tìm kiếm và đánh giá hiệu suất truy vấn ảnh thực nghiệm trên bộ ảnh COREL, Caltech256 với một cải tiến về cấu trúc KD-Tree được đề xuất trong các công trình đã công bố trước đây bởi chúng tôi gọi là CB KD-Tree. Cuối cùng, một số ưu nhược điểm trên các biến thể KD-Tree được đánh giá để có những cải tiến tiếp theo nhằm nâng cao hiệu suất truy vấn ảnh.

Từ khóa: Biến thể KD-Tree, hiệu suất truy vấn, KD-Tree, truy vấn ảnh.

1. MỞ ĐẦU

Quản lý dữ liệu ảnh số là một lĩnh vực nghiên cứu chuyên sâu về cơ sở dữ liệu đa phương tiện trong những thập niên gần đây. Dữ liệu ảnh số gia tăng theo thời gian cũng là một thách thức cho vấn đề lưu trữ và tìm kiếm. Để giải quyết vấn đề này, sự ra đời của các cấu trúc dữ liệu đa chiều, đa chỉ mục là cần thiết để nâng cao hiệu suất lưu trữ cũng như cải thiện thời gian tìm kiếm [8]. Vì vậy, cấu trúc dữ liệu đa chiều đóng một vai trò quan trọng và ảnh hưởng lớn đến hiệu suất bài toán truy vấn ảnh.

Các hệ tìm kiếm ảnh được thực hiện bởi nhiều cách tiếp cận khác nhau nhằm mang lại hiệu suất cao đáp ứng nhu cầu người dùng. Trong những thập niên gần đây, các công trình tìm kiếm ảnh mang lại hiệu suất cao phải kể đến đó là sự tích hợp của nhiều kỹ thuật học máy vào một cấu trúc dữ liệu nhằm nâng cao hiệu suất về độ chính

xác, cải thiện thời gian tìm kiếm và tăng khả năng lưu trữ để đáp ứng nhu cầu gia tăng dữ liệu ảnh số theo thời gian. Trong các phương pháp thực hiện bài toán tìm kiếm ảnh thì cấu trúc dữ liệu lưu trữ là một vấn đề được nhiều nhóm nghiên cứu quan tâm để cải tiến và phát triển. Trong đó, cấu trúc dữ liệu dạng cây có rất nhiều công trình đã công bố, đặc biệt là các biến thể của KD-Tree, R-Tree, S-Tree, v.v cùng với nhiều cải tiến khác đã được nhiều công trình công bố trong thời gian qua [8]. Trong bài báo này, chúng tôi thực hiện một khảo sát về tính hiệu quả của các hệ tra cứu ảnh dựa trên các biến thể KD-Tree nhằm minh chứng, đánh giá tính khả thi và so sánh hiệu suất cho bài toán truy vấn ảnh, đồng thời so sánh với một biến thể KD-Tree mà chúng tôi đề xuất.

2. CÁC CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN

Các nghiên cứu về cấu trúc KD-Tree cho bài toán xử lý dữ liệu đa chiều nó chung và bài toán tìm kiếm ảnh nói riêng đã có nhiều công bố với kết quả khả quan trong những thập niên vừa qua. Trên cơ sở KD-Tree nguyên thủy [1], một số biến thể KD-Tree được xây dựng và ứng dụng cho bài toán truy vấn ảnh được đánh giá là khả thi, hiệu quả và mang tính thời sự, cụ thể là:

N. Puviarasan và cộng sự (2015) [2] đã thực hiện bài toán truy vấn ảnh dựa trên cấu trúc KD-Tree để xây dựng cấu trúc Index KD-Tree nhằm lưu trữ tập ảnh có kích thước lớn, đồng thời kết hợp với các loại đặc trưng hình ảnh để đánh giá các kết quả thực nghiệm trên bộ ảnh COREL. Trong cấu trúc Index KD-Tree, tác giả quan tâm đến việc chèn nút vào cây, sau đó thực hiện quá trình tìm kiếm trên cây theo vùng không gian gần nhất với điểm truy vấn theo phương pháp lập chỉ mục. Công trình này được đánh giá là hiệu quả cho bài toán lưu trữ tập dữ liệu lớn, thời gian truy vấn nhanh và khả thi cho bài toán tìm kiếm ảnh theo nội dung.

Y.H. Sharath Kumar và cộng sự (2015) [6] đề xuất mô hình tìm kiếm ảnh theo nội dung dựa trên cấu trúc KD-Tree được đề xuất và đánh giá là hiệu quả gồm hai pha: Pha huấn luyện gồm: (1) Tập dữ liệu ảnh được phân đoạn; (2) trích xuất đặc trưng; (3) xây dựng cấu trúc Indexing KD-Tree; (4) Xây dựng hệ tri thức. Pha truy vấn dựa trên ảnh phát thảo gồm: (1) Đầu vào là một ảnh phát thảo; (2) trích xuất đặc trưng; (3) thực hiện truy vấn dựa trên hệ tri thức đã xây dựng từ cấu trúc Indexing KD-Tree, đồng thời dựa vào véc-tơ đặc trưng của hình ảnh phát thảo để đối sánh chọn ra tập kết quả truy vấn tốt nhất. Sau khi xây dựng cấu trúc chỉ mục Indexing KD-Tree là một hệ tri thức để áp dụng cho bài toán truy vấn ảnh, quá trình tìm kiếm và đối sánh được thực hiện thêm một bước lọc ra những kết quả tối ưu nhất. Đây là một ưu điểm so với các công trình tìm kiếm trực tiếp trên cấu trúc KD-Tree giúp nâng cao hiệu suất. Tuy nhiên, để xây dựng hệ tri thức này thì phải tốn một lượng chi phí nhất định.

Fengquan Zhang (2019) [4] và cộng sự đã thực hiện xây dựng cấu trúc Vocabulary-KDTree nhằm thiện 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; (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 2 nhóm: (1) nhóm chứa các đặc trưng hình ảnh; (2) nhóm các nút lá cùng với việc điều chỉnh các trọng số liên quan quá trình huấn luyện để xây dựng cây. Mô hình truy vấn ảnh được thực hiện theo hai pha: 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. 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.

Y Narasimhulu (2021) [7] đã đề xuất một mô hình phân lớp hình ảnh dựa vào KD-Tree. Trong mô hình này, từ một tập dữ liệu lớn chưa phân loại được thực hiện xây dựng tập dữ liệu Coresets trên cơ sở thuật toán đề xuất của tác giả. Sau đó dựa vào thuật toán xây dựng KD-Tree để hình thành cây KD-Tree chứa tập dữ liệu này. 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 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 ảnh dựa vào cấu trúc KD-Tree được đánh giá là khá tốt.

Sumeet Gill và cộng sự (2021) [8] đã tóm tắt các biến thể KD-Tree. Trong đó, dựa vào chức năng hay tính chất của tập dữ liệu thực nghiệm; các thao tác thực hiện trên cây mà những công trình này đề xuất kỹ thuật xây dựng biến thể tương ứng, bao gồm: **Movies** (2009) là một cấu trúc chỉ mục đa chiều để lưu trữ các đối tượng có tính chất chuyển động theo thời gian như video, yêu cầu các kỹ thuật lập chỉ mục, lưu trữ và truy vấn trên các tập dữ liệu lớn. **Movies** đáp ứng nhu cầu cập nhật dữ liệu nhiều, thời gian truy vấn thấp. Vì vậy, **Movies** xử lý tốt cho các đối tượng chuyển động bằng cách xét các ảnh chụp xung quanh đối tượng chuyển động để dự đoán tính chất và hành động của đối tượng cần khảo sát. **Randomly Projected K-d Trees** (2011) và **Squarish kd Trees** (2010) là một kỹ thuật xây dựng nhiều cây KD-Tree thành rừng ngẫu nhiên để ứng dụng cho bài toán truy vấn ảnh đa đối tượng. **Relaxed KD Tree** (2010) là một biến thể của KD-Tree, linh hoạt và thích ứng tốt cho các thao tác cập nhật dữ liệu thường xuyên trên cấu trúc KD-Tree. **Buffer k-d Trees** (2014) là một cấu trúc KD-Tree được xây dựng trên cơ sở kết hợp kỹ thuật tìm kiếm láng giềng và chú trọng bộ nhớ GPUs (Graphics Processing Units) trong xử lý hình ảnh. **Progressive k-d Tree**

(2017) là một cấu trúc cải tiến từ KD-Tree cổ điển bằng cách sử dụng thuật toán tìm kiếm láng giềng k-NN trong truy vấn dữ liệu.

Trên cơ sở các công trình truy vấn ảnh sử dụng cấu trúc KD-Tree, một số biến thể được thực hiện cho bài toán truy vấn ảnh. Mỗi công trình có những ưu và nhược điểm riêng cần cải tiến. Vì vậy, trong bài báo này một số khuyến nghị được trình bày dựa trên các biến thể KD-Tree, từ đó đề xuất một số cải tiến để nâng cao hiệu suất tìm kiếm ảnh dựa trên cấu trúc KD-Tree.

3. MỘT SỐ BIẾN THỂ KD-TREE CHO BÀI TOÁN TRUY VẤN ẢNH

Cấu trúc KD-Tree nguyên thủy được Bentley (1975) [2] đề xuất là một dạng cây tìm kiếm nhị phân, biểu diễn các điểm trong không gian đa chiều gồm một nút gốc, các nút trong và nhiều nút lá. Một nút không phải là nút lá chia không gian thành hai phần, phần các điểm ở bên trái của không gian này biểu thị bằng cây con trái và các điểm ở bên phải biểu thị bằng cây con phải. Cây KD-Tree dùng để biểu diễn dữ liệu đa chiều và được lưu trữ tại tất cả các nút trên cây. Trên cơ sở này, một số biến thể KD-Tree được xây dựng và áp dụng cho bài toán truy vấn ảnh, cụ thể là:

k-NN KD-Tree là một biến thể KD-Tree được Abdulkadhem Abdulkareem và công sự (2019) [5] đề xuất cho bài toán tìm kiếm ảnh theo nội dung dựa trên cấu trúc KD-Tree kết hợp thuật toán k-NN. Quá trình xây dựng cấu trúc này được thực hiện theo hai pha: Pha huấn luyện phân lớp dữ liệu hình ảnh sử dụng thuật toán k-NN để phân loại theo k láng giềng gần nhất. Pha phân lớp được tính toán theo khoảng cách gần nhất để tìm láng giềng của phần tử đầu vào trong tập dữ liệu huấn luyện. Vì vậy, để tăng hiệu suất tìm kiếm, sự kết hợp k-NN và cấu trúc KD-Tree để loại bỏ ngay lập tức một số điểm không thỏa điều kiện vì vùng tìm kiếm xa hơn các điểm láng giềng.

Các đặc trưng hình ảnh được sử dụng là kết cấu (*Texture*) và hình dạng (*Shape*) của đối tượng trích xuất bằng cách sử dụng mẫu nhị phân cục bộ LBP (*Local Binary Pattern*) và biểu đồ hướng cạnh EHD (*Edge Histogram Descriptor*); Nén dữ liệu bằng SVD (*Singular Value Decomposition*) được sử dụng để giảm số chiều đặc trưng của vector hình ảnh. Biến thể KD-Tree trong công trình này được sử dụng để lưu trữ dữ liệu hình ảnh tại các nút lá. Trong đó, quá trình huấn luyện tập dữ liệu phân lớp trên KD-Tree kết hợp k-NN nhằm làm giảm độ phức tạp tính toán. Cấu trúc k-NN KD-Tree được xây dựng là quá trình đệ quy bắt đầu từ nút gốc, sau đó khi thêm dữ liệu vào cây thì quá trình phân chia thành nửa cây con trái và nửa cây con phải bằng mặt phẳng phân tách từ nút gốc để tạo thành cây nhị phân cân bằng. Giá trị được chọn làm mặt phẳng chia phân vùng không gian nửa cây con là giá trị trung vị trong miền không gian của k-NN KD-Tree. Vì vậy, quá trình xây dựng k-NN KD-Tree này được tính toán giá trị trung vị và lặp lại đệ quy.

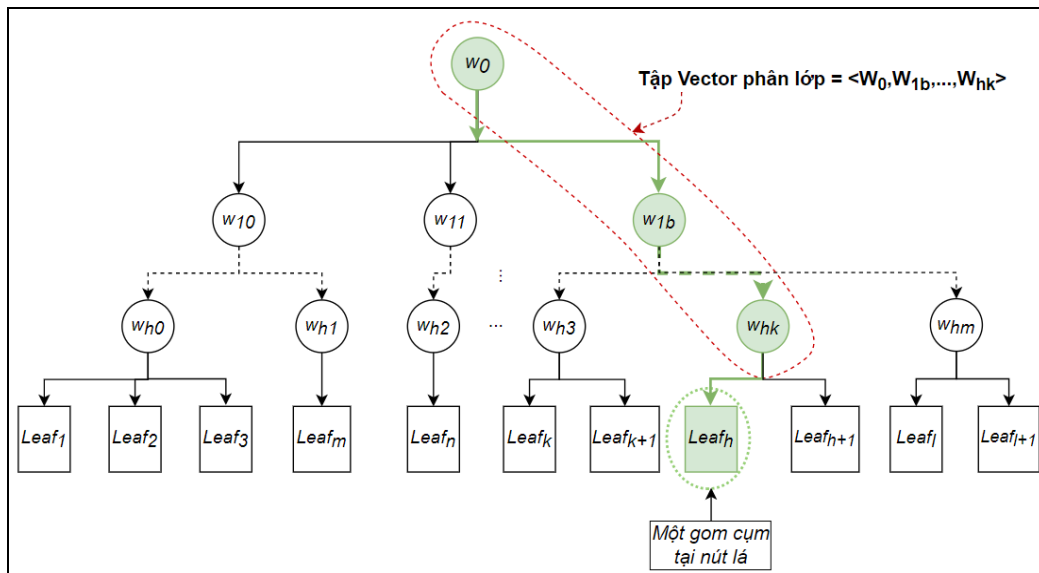
Trong công trình này, để giải quyết bài toán tốt hơn giải thuật k-NN bằng cách sử dụng cấu trúc KD-Tree kết hợp k-NN nhằm phân chia không gian đặc trưng thành các phân vùng không gian đặc trưng; khi đó quá trình tìm kiếm sẽ loại bỏ các vùng không gian không cần thiết ngay lập tức bởi vì phân vùng này xa hơn so với k láng giềng của phần tử đang tìm kiếm. Kết quả thực nghiệm được đánh giá trên tập ảnh COREL. Đầu vào cho hệ thống là một hình ảnh truy vấn và cơ sở dữ liệu ảnh; đầu ra là tập ảnh tương tự cho quá trình truy vấn ảnh. Hệ thống đề xuất được đánh giá bằng cách sử dụng các giá trị độ chính xác trong khoảng 0.43 – 0.93 và độ phủ trung bình là 0.643 theo các giá trị TopK. Các giá trị của độ chính xác trung bình là 0.7286 cho toàn bộ hệ thống. Mô hình đề xuất cho hệ thống này gồm: (1) Trích xuất đặc trưng hình ảnh; (2) Giảm chiều dữ liệu cho véc-tơ đặc trưng; (3) Lưu trữ lên cấu trúc KD-Tree; (4) Truy vấn tập ảnh tương tự theo TopK để trích xuất tập ảnh tương tự với ảnh đầu vào.

Distributed KD-tree là một biến thể KD-Tree được Mahesha D M và cộng sự (2021) [6] đề cập đến bài toán truy vấn ảnh theo nội dung. Distributed KD-tree được phát triển là cấu trúc đa nhánh, cân bằng trên cơ sở phát triển KD-Tree nhị phân. Distributed KD-tree được xây dựng trên cơ sở phân tách nút gốc thành các nhánh con và đệ quy. Giá trị phân chia nhánh con là trục có phương sai lớn nhất. Quá trình tìm kiếm trên Distributed KD-tree được thực hiện từ nút gốc đến nút lá dựa vào k láng giềng của phần tử tìm kiếm để xác định nút lá chứa tập ảnh tương tự với ảnh đầu vào. Bên cạnh đó, dữ liệu lưu trữ trên Distributed KD-tree theo phương pháp lập chỉ mục có xác định trước nên quá trình tìm kiếm nhanh, hiệu quả.

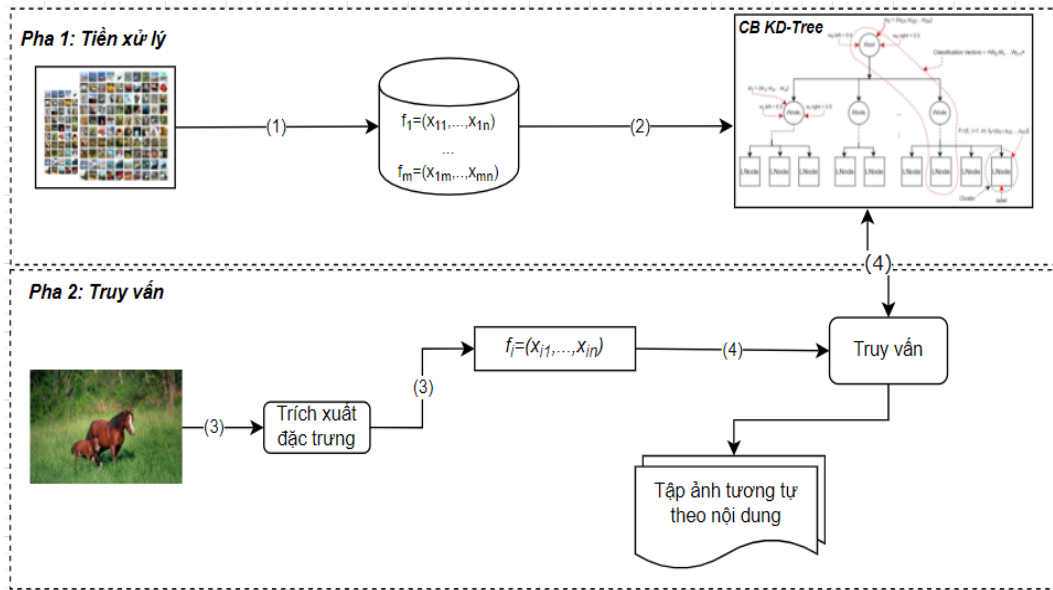
Đối tượng hình ảnh được phân đoạn sử dụng kỹ thuật hợp nhất vùng (*Region Merging Technique*). Sau đó, các đặc trưng như phép biến đổi bất biến tỷ lệ (*Scale invariant features transforms*) và biểu đồ màu được trích xuất. Cuối cùng, thuật toán PCA (*Principal Component Analysis*) được sử dụng cho giảm chiều dữ liệu để lưu trữ trên Distributed KD-tree. Kết quả thực nghiệm trên bộ ảnh Clatech256 với kết quả và vượt trội hơn đáng kể so với các phương pháp cơ sở trước đây; đồng thời có thể áp dụng cho tập dữ liệu quy mô lớn. Thực nghiệm được trình bày theo phương pháp lấy TopK kết hợp với số chiều dữ liệu được lựa chọn, đồng thời nhóm tác giả còn thực nghiệm theo cách chia tỷ lệ tập Training và Testing bằng nhiều số liệu khác nhau để đánh giá hiệu suất cho bài toán truy vấn ảnh. Mô hình đề xuất của hệ thống này gồm: (1) Tiền xử lý ảnh; (2) Trích xuất đặc trưng SIFT, HOG; (3) Giảm chiều dữ liệu bằng thuật toán PCA; (4) Xây dựng cấu trúc Distributed K-D tree để lưu trữ dữ liệu hình ảnh và trích xuất tập ảnh tương tự với ảnh đầu vào dựa trên Distributed K-D tree.

4. MÔ HÌNH TRUY VẤN ẢNH DỰA TRÊN CẤU TRÚC CB KD-TREE

Trên cơ sở cấu trúc KD-Tree nguyên thủy [1] và kết quả đã công bố trong công trình [10] trước của chúng tôi. Cấu trúc CB KD-Tree xây dựng và áp dụng cho bài toán tìm kiếm ảnh được đánh giá là khả thi và hiệu quả trên các bộ ảnh thực nghiệm. Trong công trình [10] cấu trúc CB KD-Tree được xây dựng là cây đa nhánh cân bằng, được xây dựng theo phương pháp phân lớp dữ liệu hình ảnh bằng huấn luyện tập véc-tơ phân lớp từ nút gốc đến các tầng của nút trong để hình thành phân cụm tại nút lá. Trong bài báo này, cấu trúc CB KD-Tree được kế thừa quá trình xây dựng như *hình 1*, và tìm kiếm từ công trình [10]; sau đó tiến hành thực hiện bài toán tìm kiếm ảnh theo nội dung thực nghiệm trên bộ dữ liệu COREL, Caltech256 để so sánh với kết quả truy vấn với công trình [5], [6]. Mô hình truy vấn ảnh được đề xuất như *hình 2* gồm các bước: (1) Trích xuất vector đặc trưng cho bộ ảnh thực nghiệm; (2) Lưu trữ tập véc-tơ đặc trưng hình ảnh trên CB KD-Tree; (3) Trích xuất véc-tơ đặc trưng cho ảnh cần truy vấn; (4) Tìm kiếm trên CB KD-Tree trích xuất tập ảnh tương tự với ảnh đầu vào.



Hình 1. Cấu trúc CB KD-Tree



Hình 2. Mô hình truy vấn ảnh trên CB KD-Tree

Bảng 1. So sánh đặc điểm các biến thể KD-Tree

k-NN KD-Tree [5]	Distributed KD-Tree [6]	CB KD-Tree
Cây nhị phân cân bằng. Xây dựng dựa trên KD-Tree, đệ quy quá trình tách nút gốc và lưu trữ dữ liệu tại nút lá. Pha huấn luyện phân lớp bởi thuật toán k-NN; pha phân lớp ảnh theo k láng giềng với ảnh đầu vào.	Cây đa nhánh, cân bằng. Xây dựng dựa trên KD-Tree hình thành thành cây chỉ mục. Quá trình tìm kiếm theo k láng giềng để tìm đến nút lá lưu trữ tập ảnh tương tự và lấy kết quả theo TopK.	Cây đa nhánh, cân bằng. Xây dựng theo phương pháp phân lớp dữ liệu. Phân lớp ảnh theo từng tầng từ gốc đến lá, dữ liệu lưu tại nút lá. Pha huấn luyện CB KD-Tree và pha truy vấn tìm tập ảnh tương tự tại nút lá.
Đặc trưng ảnh gồm Texture, Shape; giảm chiều đặc trưng bằng SVD.	Đặc trưng trích xuất gồm SIFT và HOG, giảm chiều đặc trưng bằng PCA	Đặc trưng trích xuất gồm SIFT, Shi, Color HOG.
Mô hình truy vấn gồm: (1) Trích xuất đặc trưng; (2) Giảm chiều dữ liệu; (3) Lưu trữ và huấn luyện k-NN KD-Tree; (4) Truy vấn trên k-NN KD-Tree	Mô hình truy vấn gồm: (1) Tiền xử lý ảnh; (2) Trích xuất đặc trưng; (3) Giảm chiều dữ liệu; (4) Lưu trữ dữ liệu trên Distributed K-D tree và tìm tập ảnh tương tự.	Mô hình truy vấn gồm: (1) Trích xuất vector đặc trưng; (2) Lưu trữ trên KD-Tree; (3) Trích xuất đặc trưng cho ảnh truy vấn; (4) Tìm kiếm trên CB KD-Tree tập ảnh tương tự.

Dữ liệu thực nghiệm COREL	Dữ liệu thực nghiệm Caltech256	Dữ liệu thực nghiệm COREL và Caltech256
---------------------------	--------------------------------	---

5. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

5.1. Dữ liệu và môi trường thực nghiệm

Trong công trình này, dữ liệu thực nghiệm trên cơ sở lấy kết quả so sánh trên cấu trúc CB KD-Tree với các công trình [5, 6] là bộ ảnh COREL và Caltech256 mô tả trong *bảng 2*. Mỗi hình ảnh được trích xuất một vector đặc trưng gồm 225 thành phần đó là: véc-tơ 225 chiều có các đặc trưng Color MPEG7 (25 giá trị), SIFT MPEG7 (25 giá trị), SHI MPEG7 (25 giá trị), Color Newton (6 giá trị), HOG (36 giá trị), Sobel HOG (36 giá trị), Laplace HOG (36 giá trị) và High Pass HOG (36 giá trị).

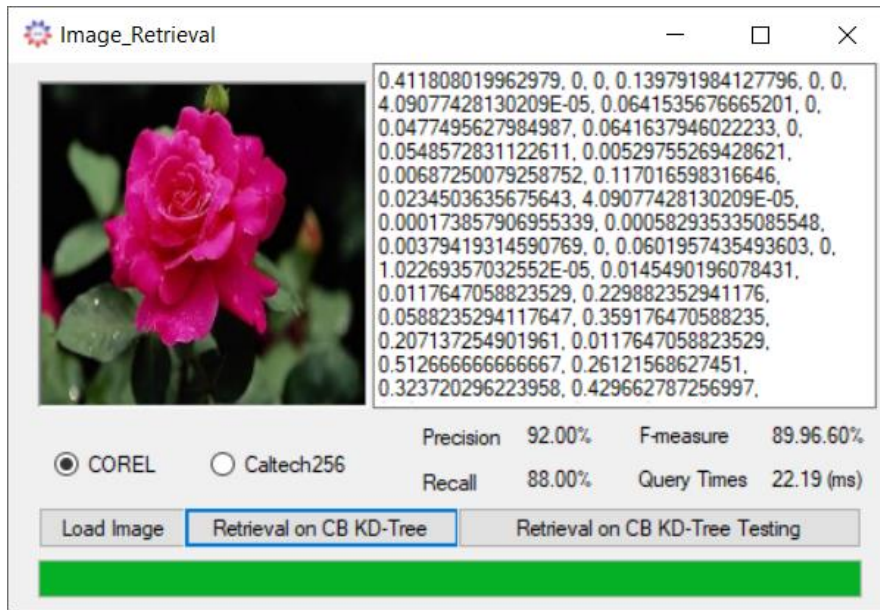
Bảng 2. Mô tả các tập dữ liệu ảnh thực nghiệm

TT	Tập ảnh	Số lượng ảnh	Số phân lớp
1	COREL	1,000	10
2	Caltech256	30,607	257

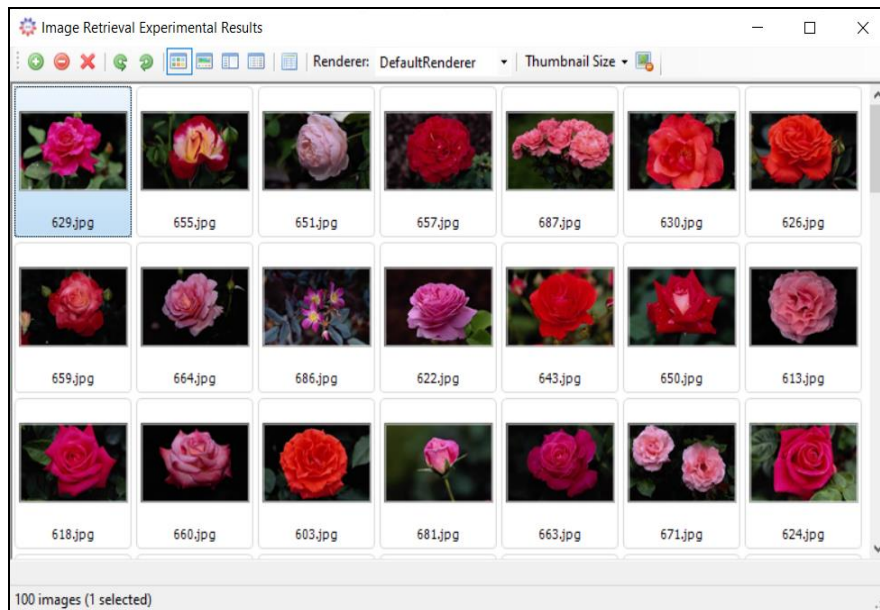
Môi trường thực nghiệm tìm kiếm ảnh tương tự **CB-KDT** được xây dựng trên nền tảng dotNET Framework 4.5, ngôn ngữ lập trình C#. Cấu hình máy tính: Intel(R) Core™ i5-5200U, CPU 2.7GHz, RAM 16GB và hệ điều hành Windows 10 Professional.

5.2. Kết quả thực nghiệm và đánh giá

Xây dựng cây CB KD-Tree minh họa bởi *hình 3*. Kết quả truy vấn ảnh 629.jpg (COREL) minh họa bởi *hình 4*. Kết quả thực nghiệm trên bộ ảnh COREL, Caltech256 trên CB KD-Tree trình bày trong *bảng 3*.



Hình 3. Hệ truy vấn CB-KDT



Hình 4. Kết quả tập ảnh tương tự với ảnh 629.jpg

Bảng 3. Hiệu suất truy vấn các bộ ảnh thực nghiệm của hệ CB-KDT

Tập ảnh	Độ chính xác trung bình	Độ phủ trung bình	Độ dung hòa trung bình
COREL	0.7808	0.6998	0.7370
Caltech256	0.7192	0.7050	0.7120

Kết quả truy vấn ảnh trên tập COREL từ công trình [5] với TopK = 100 thì độ chính xác trung bình là 0.6430; Nếu lấy trung bình của các TopK thì kết quả ở **bảng 4**.

Bảng 4. Kết quả thực nghiệm trên bộ COREL trên công trình [5]

Chủ đề	TopK=10	TopK=20	TopK=30	TopK=40	TopK=50	TopK=100	Trung bình
Beach	0.8	0.8	0.766	0.725	0.68	0.57	0.723
Bus	1.0	0.95	0.966	0.975	0.88	0.90	0.945
Castle	0.9	0.85	0.833	0.675	0.62	0.62	0.749
Dinosaur	1.0	1.0	1.0	1.0	0.98	0.93	0.985
Elephant	0.5	0.4	0.3	0.35	0.38	0.43	0.393
Flower	1.0	1.0	0.966	0.925	0.9	0.89	0.946
Horse	0.8	0.8	0.56	0.535	0.5	0.50	0.615
Meal	0.8	0.8	0.666	0.675	0.6	0.51	0.675
Mountain	0.6	0.55	0.533	0.43	0.48	0.51	0.517
Peoples	0.90	0.85	0.76	0.675	0.66	0.57	0.735
Avg. Pre	0.83	0.80	0.735	0.697	0.668	0.643	0.7286

Công trình [5] dữ liệu thực nghiệm theo TopK =100 để so sánh như *bảng 5*.

Bảng 5. So sánh hiệu suất truy vấn trên bộ ảnh COREL theo TopK=100

Chủ đề	k-NN KD-Tree [5]	CB-KDT	Chủ đề	k-NN KD-Tree [5]	CB-KDT
Beach	0.57	0.8041	Flower	0.89	0.8975
Bus	0.90	0.7210	Horse	0.50	0.6697
Castle	0.62	0.7484	Meal	0.51	0.7242
Dinosaur	0.93	0.9645	Mountain	0.51	0.7153
Elephant	0.43	0.7944	Peoples	0.57	0.7684
Trung bình	0.690	0.8065	Trung bình	0.596	0.7593

Công trình [6], thực nghiệm lấy 30 phân lớp có kết quả tốt nhất để trình bày trên bộ ảnh Caltech256 với kết quả trung bình là 0.89. Do đó, trong bài báo này, chúng tôi thực hiện lấy 30 phân lớp có kết quả truy vấn tốt nhất trên cấu trúc KD-Tree đã đề xuất, đồng thời hệ truy vấn CB-KDT chỉ thực hiện thực nghiệm trên tập véc-to đặc trưng 225 chiều gồm các loại đặc trưng SIFT, HOG và cũng chia tỷ lệ tập Training 30-50-70 như công trình [6] đã thực nghiệm. Lấy theo các giá trị TopK trên 30 chủ đề ảnh tốt nhất và chia tập Training theo các tỷ lệ là 50% và 70%. Kết quả so sánh như *bảng 5*.

Bảng 5. So sánh theo TopK trên 30 bộ ảnh thực nghiệm bộ Caltech256

TopK	Tỷ lệ Training	Distributed KD-Tree [6]	CB-KDT
TopK = 30	50%	0.9007	0.9015
TopK = 25	50%	0.8923	0.9120
TopK = 30	70%	0.8977	0.9149
TopK = 20	70%	0.8912	0.9308

Kết quả truy vấn ảnh trên cùng tập dữ liệu thực nghiệm COREL, Caltech256 với 3 cấu trúc mở rộng từ KD-Tree cho thấy kết quả trên CB KD-Tree cao hơn các biến thể k-NN KD-Tree và Distributed KD-Tree bởi các lý do sau: (1) CB KD-Tree xây dựng theo phương pháp phân lớp dữ liệu hình ảnh, có pha huấn luyện trọng số nên quá trình điều chỉnh véc-tơ phân lớp để thực hiện phân lớp trước khi gom cụm trên CB KD-Tree; (2) Quá trình phân lớp cho một đối tượng nhiều lần tại các tầng trên CB KD-Tree nên hiệu suất phân lớp cao; (3) CB KD-Tree là cây đa nhánh cân bằng, quá trình xây dựng cây có khả năng mở rộng cho số nút lá lớn, hiệu suất lưu trữ và tìm kiếm khá tốt.

6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi thực hiện một khảo sát nhằm đánh giá hiệu suất truy vấn ảnh trên các biến thể của KD-Tree. Mỗi biến thể của KD-Tree đều có những ưu và nhược điểm riêng để thích ứng với từng loại dữ liệu tiếp cận và hướng đến tập dữ liệu mục tiêu. Đồng thời so sánh kết quả theo các cách lấy TopK khác nhau để đánh giá hiệu suất các mô hình tìm kiếm ảnh được khảo sát. Hướng phát triển tiếp theo của chúng tôi là đề xuất mô hình tra cứu ảnh dựa trên phương pháp xây dựng cây KD-Tree tích hợp các kỹ thuật học máy để cải tiến hiệu suất phân lớp hình ảnh cho bài toán tìm kiếm ảnh tương tự.

LỜI CẢM ƠN

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.

TÀI LIỆU THAM KHẢO

- [1] Bentley, Jon Louis (1975). Multidimensional binary search trees used for associative searching, *Communications of the ACM*, 18,9, pp. 509-517.
- [2] Puviarasan, N., and R. Bhavani (2015). A new indexing technique to retrieve images using integration of colour-size, texture and shape features, *International Journal of Applied Pattern Recognition*, 2,3, pp. 280-302.
- [3] Kumar, YH Sharath, and N. Pavithra (2015). KD-Tree approach in sketch based image retrieval, *International Conference on Mining Intelligence and Knowledge Exploration*, Springer, Cham.
- [4] Zhang, Fengquan, Yahui Gao, and Liuqing Xu (2019). An adaptive image feature matching method using mixed Vocabulary-KD tree, *Multimedia Tools and Applications*, pp. 1-19.
- [5] Abdulkadhem, Abdulkadhem Abdulkareem, and Tawfiq A. Al-Assadi (2019). Proposed a Content-Based Image Retrieval System Based on the Shape and Texture Features, *Int. J. Innovat. Technol. Explor. Eng* 8, pp. 2189.
- [6] Mahesha, D. M., and Sharath Kumar YH (2021). A Distributed KD-Tree and Region Merging in Object Retrieval, Volume 22, Issue 11.
- [7] Narasimhulua, Y., et al (2021). Ckd-Tree: An Improved Kd-Tree Construction Algorithm.
- [8] Gill, Sumeet, and Meenakshi Hooda (2021). The Design Perspective of the Structures Based on kd Tree, *Rising Threats in Expert Applications and Solutions*, Springer, Singapore, pp. 515-524.
- [9] Nguyễn Thị Định, Thế Thành Văn, Mạnh Thành Lê (2021). Phân lớp ảnh bằng cây KD-Tree cho bài toán tìm kiếm ảnh tương tự, *Các công trình nghiên cứu, phát triển và ứng dụng Công nghệ Thông tin và Truyền thông, Tạp chí Thông tin và Truyền thông*, tr. 40-52.
- [10] Nguyễn Thị Định, Thế Thành Văn, Mạnh Thành Lê (2021), Một phương pháp phân lớp dựa trên cấu trúc KD-Tree cho bài toán tìm kiếm ảnh theo ngữ nghĩa, *Kỹ yếu hội nghị quốc gia lần thứ 14 về nghiên cứu cơ bản và ứng dụng công nghệ thông tin*, Fair2021, TP. Hồ Chí Minh, 23/12/2021. DOI: [10.15625/vap.2021.0075](https://doi.org/10.15625/vap.2021.0075)

AN ASSESSMENT OF IMAGE RETRIEVAL PERFORMANCE ON KD-TREE VARIES

Nguyen Thi Dinh^{1,3*}, Van The Thanh², Le Manh Thanh¹

¹ Faculty of Information Technology, University of Sciences, Hue University

² HCMC University of Education

³ Faculty of Information Technology, University of Food Industry HCMC

*Email: dinhnt@hufi.edu.vn

ABSTRACT

In this paper, a survey of evaluating image retrieval performance on some KD-Tree variants is presented. There are many KD-Tree variants used for multidimensional spatial data processing problems including digital image data such as Randomly Projected K-d Trees, Buffer k-d Trees, Progressive k-d Trees, etc. In which, Distributed KD-Tree and k-NN KD-Tree are two KD-Tree variants that have been investigated on construction techniques, search methods, and performance evaluation of experimental image retrieval on COREL, Caltech256 image data sets. An improvement on the KD-Tree structure proposed in previously published works by us. Finally, some advantages and disadvantages of KD-Tree variants are evaluated to have improvements image retrieval performance for next paper.

Keywords: KD-Tree variant, Retrieval performance, KD-Tree, Image retrieval.



Nguyễn Thị Định sinh năm 1983. Bà tốt nghiệp ngành Sư phạm tin học Trường Đại học Sư phạm TP.HCM vào năm 2006, nhận bằng Thạc sĩ ngành Truyền số liệu và mạng máy tính tại Học viện Công nghệ Bưu chính viễn thông TP.HCM vào năm 2011. Hiện là nghiên cứu sinh ngành Khoa học máy tính Trường Đại học khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: xử lý ảnh, tìm kiếm ảnh và cơ sở dữ liệu.



Lê Mạnh Thạnh sinh năm 1953. Ông nhận bằng Tiến sĩ ngành khoa học máy tính tại Đại học Budapest (ELTE), Hungary vào năm 1994. Nhận học hàm Phó giáo sư tại trường Đại học Khoa học, Đại học Huế, Việt Nam vào năm 2004.

Lĩnh vực nghiên cứu: cơ sở dữ liệu, cơ sở tri thức và lập trình logic.



Văn Thế Thành sinh năm 1979. Ông tốt nghiệp chuyên ngành Toán tin tại Đại học Khoa học Tự nhiên - Đại học Quốc gia TP.HCM vào năm 2001, nhận bằng Thạc sĩ Khoa học Máy tính tại Đại học Quốc gia TP.HCM vào năm 2008. Năm 2016, nhận bằng Tiến sĩ Khoa học Máy tính tại trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: xử lý ảnh, khai thác dữ liệu ảnh và tìm kiếm ảnh.