BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

NGUYỄN MINH HẢI

PHÁT TRIỂN THUẬT TOÁN TRA CỨU ẢNH DỰA TRÊN NGỮ NGHĨA

LUẬN ÁN TIẾN SĨ NGÀNH MÁY TÍNH

TP. Hồ Chí Minh - 2024

BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

NGUYỄN MINH HẢI

PHÁT TRIỂN THUẬT TOÁN TRA CỨU ẢNH DỰA TRÊN NGỮ NGHĨA

LUẬN ÁN TIẾN SĨ NGÀNH MÁY TÍNH

Ngành: Khoa học máy tính

Mã số: 9 48 01 01

Xác nhận của Học viện Khoa học và Công nghệ	Người hướng dẫn 1	Người hướng dẫn 2
	PGS. TS. Trần Văn Lăng	TS. Văn Thế Thành

TP. Hồ Chí Minh - 2024

LỜI CAM ĐOAN

Tôi xin cam đoan luận án: "Phát triển thuật toán tra cứu ảnh dựa trên ngữ nghĩa" là công trình nghiên cứu của chính mình dưới sự hướng dẫn khoa học của tập thể hướng dẫn. Luận án sử dụng thông tin trích dẫn từ nhiều nguồn tham khảo khác nhau và các thông tin trích dẫn được ghi rõ nguồn gốc. Các kết quả nghiên cứu của tôi được công bố chung với các tác giả khác đã được sự nhất trí của đồng tác giả khi đưa vào luận án. Các số liệu, kết quả được trình bày trong luận án là hoàn toàn trung thực và chưa từng được công bố trong bất kỳ một công trình nào khác ngoài các công trình công bố của tác giả. Luận án được hoàn thành trong thời gian tôi là nghiên cứu sinh của Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Tác giả luận án

NCS. Nguyễn Minh Hải

LỜI CẢM ƠN

Luận án tiến sĩ được hoàn thiện bởi sự cố gắng của chính bản thân cùng với sự giúp đỡ tận tình của hai Thầy hướng dẫn khoa học, một số chuyên gia, đồng nghiệp, bạn bè và người thân trong gia đình.

Trước tiên, tôi xin được bày tỏ lòng biết ơn chân thành đến hai Thầy hướng dẫn khoa học PGS. TS. Trần Văn Lăng và TS. Văn Thế Thành. Nghiên cứu sinh đã nhận được những định hướng khoa học, những bài học quý báu, sự hướng dẫn tận tình và kinh nghiệm nghiên cứu khoa học quý giá trong nghiên cứu.

Tôi xin chân thành cảm ơn Ban lãnh đạo, phòng Đào tạo, các phòng chức năng của Học viện Khoa học và Công nghệ, Viện Cơ học và Tin học ứng dụng thuộc Viện Hàn lâm Khoa học và Công nghệ Việt Nam đã tạo điều kiện thuận lợi trong suốt quá trình nghiên cứu và thực hiện luận án.

Tôi xin chân thành cảm ơn tới Ban giám hiệu, Ban lãnh đạo Khoa, các đồng nghiệp là cán bộ, giảng viên Trường Đại học Sư phạm TP. Hồ Chí Minh đã quan tâm, giúp đỡ tôi hoàn thành nhiệm vụ.

Cuối cùng, tôi xin bày tỏ lòng biết ơn vô hạn tới mọi thành viên trong gia đình, sự khuyến khích động viên của gia đình là động lực để tôi hoàn thành luận án này.

Tác giả luận án

NCS. Nguyễn Minh Hải

MỤC LỤC

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

Ký hiệu	Diễn giải tiếng Anh	Diễn giải tiếng Việt
ARP	Average Retrieval Precision	Độ chính xác trung bình
AUC	Area Under the Curve	Diện tích dưới đường cong
CBIR	Content-Based Image Retrieval	Tìm kiếm ảnh theo nội dung
CDH	Color Difference Histogram	Biểu đồ chênh lệch màu
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập
CSD	Color Structure Descriptor	Bộ mô tả cấu trúc màu
GP-Tree	Growing Partitioning Tree	Cây phân hoạch tăng trưởng
DCD	Dominant Color Descriptor	Bộ mô tả màu chủ đạo
DoG	Difference of Gaussian	Đạo hàm Gauss
DNN	Deep neural Networks	Mạng nơ-ron sâu
DWT	Discrete Wavelet Transform	Phép biến đổi Wavelet rời rạc
EC	Element Center	Phần tử trọng tâm
ED	Element Data	Phần tử dữ liệu
EDH	Edge Histogram Descriptor	Bộ mô tả biểu đồ biên
FGIR	Fine-Grained Image Retrieval	Tìm kiếm ảnh chi tiết
GLCM	Gray-level co-occurrence matrix	Ma trận đồng xuất hiện mức xám
GMM	Gaussian Mixture Models	Mô hình hỗn hợp Gauss
Graph-GPTree	Neighbor Graph on GP-Tree	Đồ thị cụm lân cận trên GP-Tree
GrSOM	Graph-Self Organizing Map	Mô hình kết hợp đồ thị cụm lân cận và bản đồ tự tổ chức
HOG	Histograms of Oriented Gradients	Biểu đồ định hướng Gradient
LBP	Local Binary Patterns	Mẫu nhị phân cục bộ
LoG	Laplace of Gaussian	Phép biến đổi Laplace Gauss
LPP	Locality-Preserving Projection	Phép chiếu bảo toàn cục bộ

MAP	Mean Average Precision	Độ chính xác trung bình
MPL	Multi-layer Perceptron	Mạng perceptron nhiều lớp
OntoSBIR	Semantic-Based Image Retrieval on ontology	Tìm kiếm ảnh theo ngữ nghĩa dựa trên ontology
ORB	Oriented Fast and Rotated BRIEF	Đặc trưng định hướng và xoay vòng nhanh
OWL	Web Ontology Language	Ngôn ngữ ontology web
RDF	Resource Description Framework	Khung mô tả tài nguyên
ROC	Receiver Operating Characteristic	Đồ thị đặc tính
RF	Relevance Feedback	Phương pháp phản hồi liên quan
RNN	Recurrent Neural Networks	Mạng nơ-ron hồi quy
SBIR	Semantic-Based Image Retrieval	Tìm ảnh theo ngữ nghĩa
SBIR_GPT	Semantic-Based Image Retrieval on GP-Tree	Tìm ảnh theo ngữ nghĩa dựa trên cây GP-Tree
SBIR-grGP	Semantic-Based Image Retrieval on Graph-GPTree	Tìm ảnh theo ngữ nghĩa dựa trên Graph-GPTree
SBIR-SgGP	Semantic-Based Image Retrieval on SgGP-Tree	Tìm ảnh theo ngữ nghĩa dựa trên SgGP-Tree
SgGP-Tree	SOM-Graph-GPTree	Mô hình kết hợp mạng SOM, đồ thị cụm lân cận và GP-Tree
SDCD	Spatial Dominant Color Descriptor	Bộ mô tả màu trội không gian
SIFT	Scale Invariant Features Transform	Đặt trưng hình ảnh SIFT
SOFM	Self-Organized Feature Map	Bản đồ đặc trưng tự tổ chức
SOM	Self Organizing Map	Bản đồ tự tổ chức
SURF	Speeded Up Robust Feature	Đặc trưng hình ảnh SURF
SVM	Support Véc-tơ Machine	Máy véc-tơ hỗ trợ
TBIR	Text-Based Image Retrieval	Tìm kiếm ảnh dựa trên văn bản
WWW	World Wide Web	Mạng toàn cầu WWW

DANH MỤC BẢNG BIỂU

DANH MỤC HÌNH ẢNH

MỞ ĐẦU

1. Giới thiệu

Sự phát minh của máy ảnh kỹ thuật số đã mang lại cho con người khả năng chụp lại thế giới xung quanh và dễ dàng chia sẻ ảnh với nhau [1]. Việc chia sẻ chúng hiệu quả thường gặp khó khăn do các hạn chế trong cơ chế tìm kiếm và khám phá [2], đặc biệt là đối với những hình ảnh khó tự động xử lý hoặc lập chỉ mục. Tìm kiếm ảnh trên Web hiện nay phụ thuộc nhiều vào các thẻ tag không chính xác hoặc thiếu sót, nên nhiều hình ảnh không cung cấp gợi ý ngữ nghĩa đầy đủ, hạn chế việc tìm kiếm và khai phá chúng. Mặc dù việc tìm kiếm ảnh theo nội dung (CBIR) dựa trên các đặc trưng cấp thấp như màu sắc, diện tích, kết cấu được trích xuất từ hình ảnh đã trở nên khá tốt, tuy nhiên người dùng thường quan tâm hơn đến các khái niệm ngữ nghĩa đằng sau hoặc bên trong hình ảnh. Tìm kiếm chỉ dựa trên các đặc trưng cấp thấp sẽ không thể đáp ứng yêu cầu của người dùng. Để tăng độ chính xác của việc tìm kiếm ảnh trên Web, cần phải làm phong phú thêm chỉ mục khái niệm và ý nghĩa ngữ nghĩa của hình ảnh cũng như khắc phục về khoảng cách ngữ nghĩa. Trong tìm kiếm ảnh theo ngữ nghĩa (SBIR), ý nghĩa của ảnh có thể ở các mức độ khác nhau, có thể xếp thành ba mức độ của việc tìm kiếm thông tin hình ảnh [3]: (1) Mức độ một dựa trên các đặc trưng cấp thấp của các yếu tố hình ảnh hoặc sự kết hợp của chúng; (2) Mức độ hai gồm việc tìm kiếm ảnh theo các thuộc tính dẫn xuất hoặc nội dung ngữ nghĩa và tương ứng với mức mô tả. Các yêu cầu tìm kiếm ở mức độ này bao gồm tìm kiếm các đối tượng thuộc một loại hay lớp cụ thể; (3) Mức độ ba gồm các yêu cầu tìm kiếm theo các thuộc tính trừu tượng như các sự kiện hoặc các loại hoạt động, các bức ảnh có ý nghĩa cảm xúc.

Luận án tập trung vào việc phát triển các thuật toán tìm kiếm ảnh để nâng cao độ chính xác và hiệu quả của quá trình lập chỉ mục và tìm kiếm ảnh theo mức độ hai. Các phương pháp và kỹ thuật sau đây đã được đề xuất và thảo luận trong luận án bao gồm:

- Đo lường sự tương đồng ảnh dựa trên ngữ cảnh và không gian ngữ nghĩa.
- Đề xuất lập chỉ mục cho tập dữ liệu ảnh dựa trên cấu trúc cây phân cụm tăng trưởng GP-Tree (Growing Partitioning Tree).

- Cải tiến cấu trúc cây GP-Tree bằng cách kết hợp đồ thị và mạng SOM (Self Organizing Map), tạo ra các cấu trúc biến thể, gồm: cấu trúc đồ thị cụm lân cận trên GP-Tree (Graph-GPTree), đồ thị cụm lân cận và mạng SOM (SgGP-Tree) nhằm tăng độ chính xác của tập ảnh được trích xuất.
- Làm giàu ngữ nghĩa cho khung ontology để tăng độ chính xác của việc tìm kiếm ảnh dựa trên ngữ nghĩa.

2. Tính cấp thiết của luận án

Sự tăng trưởng mạnh mẽ trong việc sử dụng thiết bị điện tử, internet và phương tiện truyền thông xã hội trong mọi khía cạnh của cuộc sống hàng ngày đã dẫn đến việc tạo ra một lượng dữ liệu khổng lồ. Nghiên cứu của EMC/IDC Digital Universe năm 2018 dự đoán rằng đến năm 2025, lượng dữ liệu được tạo ra hàng năm trên toàn thế giới sẽ đạt 175 zettabyte (175 nghìn tỷ gigabyte) [4]. Dữ liệu này chủ yếu bao gồm các tập tin đa phương tiện dưới dạng hình ảnh, video và âm thanh. Ví dụ như trang web Flickr có tốc độ tải lên hình ảnh hàng ngày lên đến khoảng 4,5 triệu ảnh, và kho lưu trữ hình ảnh của Facebook chứa hơn 300 triệu hình ảnh mà người dùng đã tải lên. Mặc dù các hình ảnh được lưu trữ trong các kho lưu trữ lớn như vậy chủ yếu tập trung vào việc lưu trữ và hiển thị, việc trích xuất thông tin từ các kho lưu trữ đa phương tiện này có thể được tối ưu hóa thông qua việc tích hợp vào các hệ thống tìm kiếm và truy xuất dữ liệu [5].

Lượng thông tin hữu ích trong dữ liệu được tạo ra dự kiến sẽ tiếp tục tăng, và việc trích xuất thông tin quan trọng từ lượng dữ liệu lớn trở thành vấn đề quan trọng trong lĩnh vực phân tích dữ liệu lớn. Thông tin thu được từ quá trình này có giá trị trong việc đưa ra các quyết định trong nhiều lĩnh vực, bao gồm dịch vụ di động, bán lẻ, sản xuất, dịch vụ tài chính, khoa học đời sống và khoa học vật lý [6, 7]. Để quản lý hiệu quả thông tin hình ảnh ghi lại bằng các thiết bị thu thập hình ảnh kỹ thuật số như máy ảnh, hệ thống hình ảnh âm thanh (siêu âm), hình ảnh điện tử (kính hiển vi điện tử) và đồ họa máy tính, việc khai thác thông tin này cần được thực hiện một cách thông minh. Các hệ thống tìm kiếm ảnh hiện tại, chẳng hạn như Bing và Google, sử dụng một số mô tả văn bản xung quanh do con người cung cấp để suy ra ngữ nghĩa [8]. Những kỹ thuật này bỏ qua các đặc trưng hình ảnh có ý nghĩa có thể được trích xuất thông qua phân tích xử lý hình ảnh.

Ngoài ra, chú thích thủ công là không thể đối với một cơ sở dữ liệu động lớn và không thể diễn đạt chính xác nội dung và khái niệm của một hình ảnh. Trong lĩnh vực nghiên cứu khoa học máy tính, tìm kiếm dựa trên nội dung hình ảnh (CBIR) đã xuất hiện như một giải pháp hữu hiệu. CBIR cho phép tìm kiếm các hình ảnh tương ứng [9-11] đáp ứng các điều kiện truy vấn trong cơ sở dữ liệu hình ảnh dựa trên các đặc trưng hình ảnh mục tiêu như thông tin pixel, màu sắc, kết cấu, hình dạng, vv. có trong chính hình ảnh đó, mà không cần chú thích thủ công. Tuy nhiên, việc phân tích ngữ nghĩa và mô tả nội dung hình ảnh bằng ngữ nghĩa cấp cao không được thực hiện bởi CBIR, dẫn đến hiệu suất tìm kiếm vẫn chưa đáp ứng được yêu cầu của người dùng. Vì vậy, bài toán SBIR là một vấn đề được quan tâm rất nhiều bởi nhiều nhà nghiên cứu. Luận án tập trung phát triển một phương pháp SBIR hiệu quả dựa trên ngữ nghĩa.

3. Mục tiêu nghiên cứu

Mục tiêu nghiên cứu của luận án tập trung các vấn đề sau:

(1) Phân tích và rút trích các đặc trưng ngữ nghĩa từ hình ảnh.

(2) Hiểu và xử lý các thành phần trong hình ảnh để tạo ra một ngữ nghĩa cho ảnh.

(3) Phát triển một hệ thống tìm kiếm và phân loại ảnh có khả năng hiểu và đáp ứng yêu cầu người dùng một cách hiệu quả.

4. Phương pháp nghiên cứu

❖ Phương pháp lý thuyết

Tổng hợp các công trình liên quan đến tìm kiếm ảnh theo ngữ nghĩa sử dụng các phương pháp học máy và các cấu trúc lưu trữ dạng cây. Phân tích các ưu và nhược điểm của các công trình; nghiên cứu phương pháp làm giàu Ontology và phát triển mô hình tìm kiếm ảnh theo ngữ nghĩa dựa trên Ontology

Đề xuất các mô hình tìm kiếm ảnh dựa trên ngữ nghĩa; đánh giá thực nghiệm dựa trên mô hình đề xuất từ đó so sánh độ chính xác tìm kiếm ảnh với các công trình trong cùng lĩnh vực để có sự điều chỉnh và cải tiến phù hợp

❖ Phương pháp thực nghiệm

Dựa trên các phương pháp và mô hình được đề xuất trong luận án, việc cài đặt chương trình thực nghiệm được triển khai trên máy có cùng cấu hình. Dữ liệu thực nghiệm được chọn là các bộ dữ liệu ảnh đáng tin cậy, được công bố rộng rãi và đã được sử dụng trong nhiều nghiên cứu trước đó để so sánh với các kết quả thực nghiệm từ các mô hình được đề xuất để minh chứng tính đúng đắn và hiệu quả của cơ sở lý thuyết.

5. Đối tượng và phạm vi nghiên cứu

❖ **Đối tượng nghiên cứu:**

- Thuật toán tìm kiếm: Phương pháp và kỹ thuật phát triển thuật toán để tìm kiếm và phân loại (phân cụm và phân lớp) ảnh dựa trên thông tin ngữ nghĩa thay vì chỉ dựa vào các đặc trưng hình ảnh như màu sắc, hình dạng. Đồng thời nghiên cứu một số phương pháp học máy nhằm cải tiến hiệu quả tìm kiếm ảnh
- Ngữ nghĩa trong hình ảnh: Các yếu tố ngữ nghĩa cần được hiểu và xác định trong ảnh để hỗ trợ việc tìm kiếm chính xác.
- Dữ liệu hình ảnh: Các bộ dữ liệu hình ảnh có chứa thông tin ngữ nghĩa cần thiết để huấn luyện và đánh giá thuật toán.

❖ **Phạm vi nghiên cứu:**

- Phạm vi ngữ nghĩa: Nhận diện một số loại ngữ nghĩa nhất định như vật thể (ví dụ: xe, người, động vật), hành động (ví dụ: chạy, nhảy), hoặc bối cảnh (ví dụ: ngoài trời, trong nhà) bằng cách dùng ontology và ngôn ngữ truy vấn SPARQL
- Tập trung vào một hoặc một số bộ dữ liệu hình ảnh có chứa thông tin ngữ nghĩa cần thiết để huấn luyện và đánh giá thuật toán như bộ dữ liệu chuẩn như Wang [12], MS-COCO [13], ImageCLEF [17].
- Giới hạn phương pháp: Tập trung vào phương pháp dựa trên ontology để liên kết ngữ nghĩa và hình ảnh.

6. Các đóng góp của luận án

Phát triển thuật toán tìm kiếm ảnh theo ngữ nghĩa dựa trên các đặc trưng thị giác của ảnh dựa trên cấu trúc dữ liệu GP-Tree nhằm nâng cao độ chính xác tìm kiếm ảnh, gồm:

(1) Xây dựng cấu trúc dữ liệu phân cụm phân cấp GP-Tree nhằm tổ chức lưu trữ các véc-tơ đặc trưng của ảnh.

(2) Phát triển cấu trúc GP-Tree dựa trên các thuật toán học có giám sát và bán giám sát nhằm tăng hiệu quả tìm kiếm ảnh.

(3) Xây dựng hệ tìm kiếm ảnh theo ngữ nghĩa dựa trên cấu trúc GP-Tree và ontology nhằm minh chứng hiệu quả tìm kiếm ảnh của các phương pháp đề xuất.

7. Nội dung và bố cục của luận án

Cấu trúc của luận án bao gồm:

- **Chương 1:** Trình bày tổng quan về bài toán tìm kiếm ảnh, với hai hướng chính là tìm kiếm ảnh theo nội dung và tìm kiếm ảnh theo ngữ nghĩa. Các công trình nghiên cứu liên quan đã được khảo sát và phân tích nhằm xác định thách thức và hạn chế trong các phương pháp hiện có, từ đó đưa ra định hướng nghiên cứu cụ thể của luận án để khắc phục những hạn chế này. Ngoài ra, phần này cũng trình bày chi tiết các phương pháp tổ chức thực nghiệm, bao gồm việc thiết lập môi trường, lựa chọn và sử dụng tập dữ liệu, cùng các tiêu chí đánh giá hiệu suất tìm kiếm.
- **Chương 2**: Trình bày các nghiên cứu liên quan đến việc sử dụng cấu trúc cây để lưu trữ và lập chỉ mục cho tập dữ liệu ảnh. Cấu trúc cây phân cụm GP-Tree được mô tả chi tiết, bao gồm các thao tác thêm, sửa, tách, và xóa phần tử. Một mô hình tìm kiếm ảnh theo ngữ nghĩa trên cây GP-Tree dựa trên ontology đã được đề xuất, với mục tiêu cải thiện hiệu suất và độ chính xác của việc tìm kiếm. Hệ thống tìm kiếm ảnh này đã được thử nghiệm trên các bộ dữ liệu phổ biến như Wang, MS-COCO và ImageCLEF, nhằm đánh giá kết quả và tính hiệu quả của mô hình đề xuất.
- **Chương 3:** Trình bày các phương pháp cải tiến cấu trúc cây phân cụm GP-Tree nhằm nâng cao hiệu quả tìm kiếm ảnh. Cụ thể, các phương pháp như đồ thị cụm Graph-GPTree và mạng kết hợp SgGP-Tree được giới thiệu để cải thiện khả năng lưu trữ và tìm kiếm các phần tử tương tự. Ngoài ra, phương pháp tìm kiếm ảnh theo ngữ nghĩa dựa trên ontology cũng được thảo luận, với cấu trúc SgGP-Tree được sử

dụng để phân lớp đối tượng trên ảnh một cách chính xác hơn. Một mô hình tìm kiếm ảnh theo ngữ nghĩa, kết hợp giữa ontology và cấu trúc SgGP-Tree, đã được đề xuất và thử nghiệm trên các bộ dữ liệu phổ biến như Wang, MS-COCO, và ImageCLEF nhằm đánh giá hiệu quả của mô hình.

- **Kết luận và hướng phát triển:** Trình bày những kết quả đạt được định hướng phát triển tiếp theo của luận án.
- **Danh mục công trình của tác giả:** Liệt kê các công trình mà tác giả đã công bố trong quá trình thực hiện luận án.
- **Tài liệu tham khảo:** Liệt kê các tài liệu mà luận án đã tham khảo.

CHƯƠNG 1. TỔNG QUAN TÌM KIẾM ẢNH

Chương này trình bày tổng quan về bài toán tìm kiếm ảnh, với hai hướng chính là tìm kiếm ảnh theo nội dung và tìm kiếm ảnh theo ngữ nghĩa. Các công trình nghiên cứu liên quan đã được khảo sát và phân tích nhằm xác định thách thức và hạn chế trong các phương pháp hiện có, từ đó đưa ra định hướng nghiên cứu cụ thể của luận án để khắc phục những hạn chế này. Ngoài ra, phần này cũng trình bày chi tiết các phương pháp tổ chức thực nghiệm, bao gồm việc thiết lập môi trường, lựa chọn và sử dụng tập dữ liệu, cùng các tiêu chí đánh giá hiệu suất tìm kiếm.

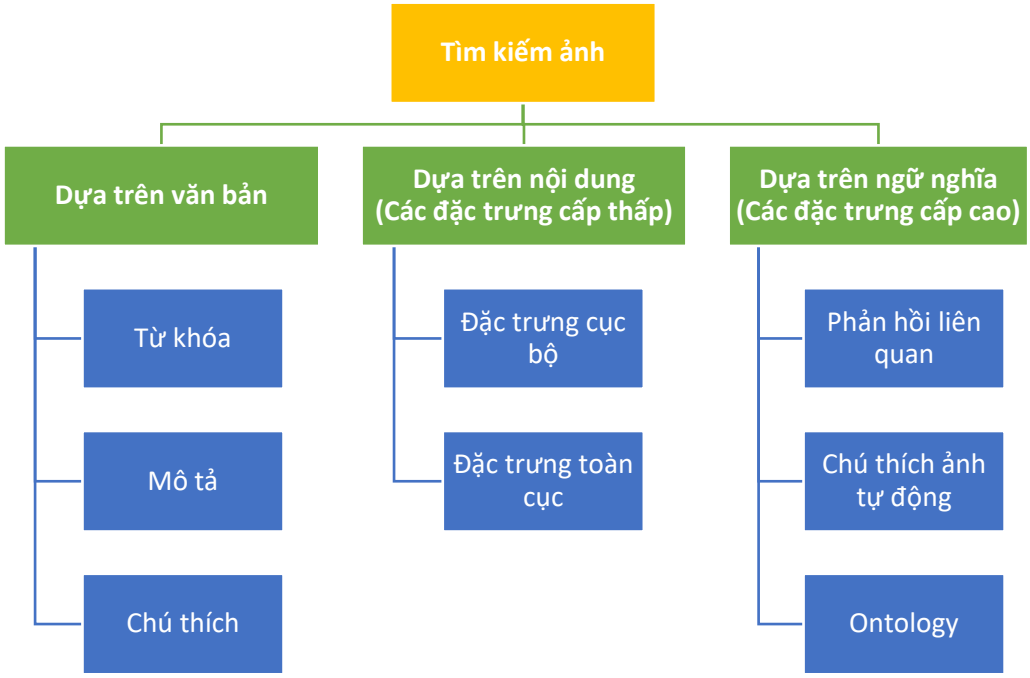
1.1. Tổng quan về tìm kiếm ảnh

Bài toán tìm kiếm ảnh trong luận án được xác định là tìm ra tập ảnh có độ tương tự gần nhất với ảnh đầu vào dựa trên độ đo tương tự giữa các ảnh [14]. Một số thuật ngữ tương tự khái niệm này xuất phát từ thuật ngữ tiếng anh “retrieval” là “truy vấn”, “tra cứu”, “truy hồi”. Tuy nhiên, thuật ngữ “tìm kiếm ảnh” sẽ được sử dụng trong luận án này để phù hợp với thuật ngữ đã được sử dụng trong các công trình nghiên cứu liên quan và không ảnh hưởng đến nội dung.

Trong những năm gần đây, bài toán tìm kiếm ảnh đã thu hút sự quan tâm lớn từ cộng đồng nghiên cứu. Tuy nhiên, vẫn còn nhiều thách thức lớn cần vượt qua như xử lý khối lượng dữ liệu hình ảnh lớn, nhận diện và phân loại hình ảnh chính xác, xử lý đa dạng ngữ cảnh và ngữ nghĩa, đảm bảo hiệu suất tìm kiếm ảnh nhanh chóng và hiệu quả. Một trong những vấn đề chính là khó khăn trong việc định vị một hình ảnh mong muốn trong một bộ sưu tập lớn và đa dạng. Mặc dù việc xác định một hình ảnh mong muốn từ một bộ sưu tập nhỏ hoàn toàn khả thi, nhưng cần có các kỹ thuật hiệu quả hơn với các bộ sưu tập chứa hàng nghìn mục. Tìm kiếm ảnh thu hút sự quan tâm của các nhà nghiên cứu trong các lĩnh vực xử lý hình ảnh, đa phương tiện, thư viện kỹ thuật số, cảm biến từ xa, thiên văn học, ứng dụng cơ sở dữ liệu và các lĩnh vực liên quan khác [15].

Tìm kiếm ảnh đã là một lĩnh vực nghiên cứu rất tích cực kể từ những năm 1970, với sự thúc đẩy của hai cộng đồng nghiên cứu chính: quản lý cơ sở dữ liệu và thị giác máy tính [16]. Do đó, tìm kiếm ảnh có thể được định nghĩa là nhiệm vụ tìm kiếm ảnh trong cơ sở

dữ liệu hình ảnh. Như mô tả trong **Hình 1.1**, các kỹ thuật tìm kiếm ảnh có thể được phân thành ba loại: tìm kiếm ảnh dựa trên văn bản (TBIR), tìm kiếm ảnh dựa trên nội dung (CBIR) và tìm kiếm ảnh dựa trên ngữ nghĩa (SBIR).

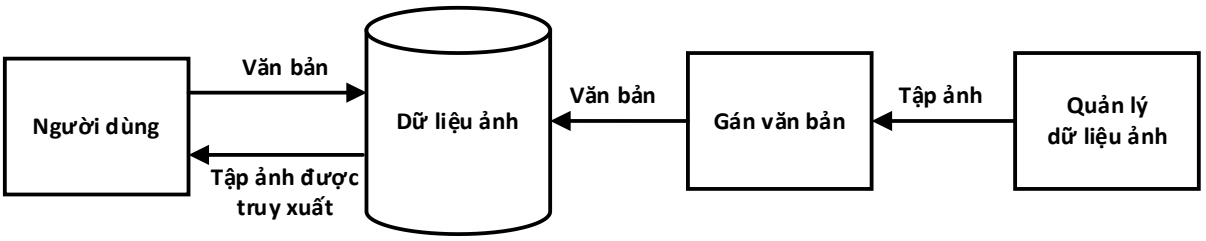


Hình 1.1. Các loại tìm kiếm ảnh.

❖ ***Tìm kiếm ảnh dựa trên văn bản (TBIR)***

Nền tảng phổ biến của TBIR đầu tiên là chú thích hình ảnh bằng văn bản và sau đó sử dụng các hệ thống quản lý cơ sở dữ liệu dựa trên văn bản để thực hiện tìm kiếm ảnh. TBIR được sử dụng để chú thích thủ công hình ảnh trong cơ sở dữ liệu bằng các chú thích, từ khóa hoặc mô tả. Quá trình này được sử dụng để mô tả cả nội dung hình ảnh và siêu dữ liệu khác của hình ảnh như: tên tệp hình ảnh, định dạng hình ảnh và kích thước hình ảnh. Sau đó, người dùng xây dựng các truy vấn dạng văn bản hoặc số để truy xuất tất cả hình ảnh đáp ứng một số tiêu chí dựa trên các chú thích này. **Hình 1.2** mô tả sơ đồ luồng điển hình của quá trình xử lý truy xuất hình ảnh. Hệ thống TBIR sử dụng hình ảnh từ cơ sở dữ liệu đã được làm giàu bằng chú thích, từ khóa hoặc mô tả. Sau đó, các yếu tố này được sử dụng để tương quan với văn bản đầu vào do người dùng cung cấp. Tuy nhiên, có một số nhược điểm trong TBIR [16]. Nhược điểm đầu tiên là các chú thích mô tả thường phải được nhập thủ công và việc chú thích này cho tập hình ảnh lớn

là không thực tế. Nhược điểm thứ hai là hầu hết các hình ảnh đều rất phong phú về nội dung và có nhiều chi tiết hơn. Người chú thích có thể đưa ra các mô tả khác nhau cho các hình ảnh có nội dung trực quan tương tự. Ngoài ra, chú thích văn bản phụ thuộc vào ngôn ngữ người dùng [4].

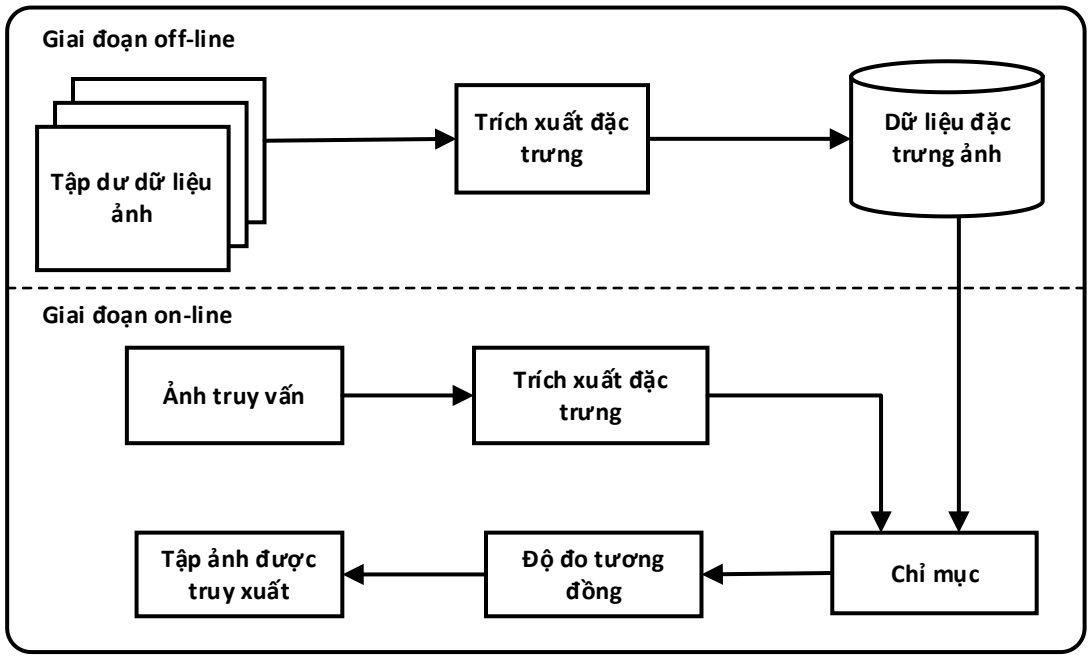


Hình 1.2. Hệ thống tìm kiếm ảnh dựa trên văn bản

❖ ***Tìm kiếm ảnh dựa trên nội dung (CBIR)***

CBIR được coi là một lĩnh vực nghiên cứu năng động và phát triển nhanh chóng. CBIR cũng được gọi là truy vấn theo nội dung hình ảnh (QBIC) và truy xuất thông tin hình ảnh dựa trên nội dung (CBVIR). Thuật ngữ CBIR bắt nguồn từ công trình của Kato [17] để tự động tìm kiếm ảnh từ cơ sở dữ liệu dựa trên màu sắc và hình dạng. Sau đó, thuật ngữ CBIR đã được sử dụng rộng rãi để mô tả quá trình tìm kiếm ảnh mong muốn từ một bộ sưu tập cơ sở dữ liệu lớn dựa trên nội dung hình ảnh trực quan được gọi là các đặc trưng (màu sắc, hình dạng, kết cấu…). Vào đầu những năm 1990, nhờ những tiến bộ của internet và các kỹ thuật sản xuất ảnh kỹ thuật số, một lượng lớn hình ảnh kỹ thuật số cho người dùng được tạo ra trong khoa học, giáo dục, y học, công nghiệp và các lĩnh vực khác. Điều này khiến những hạn chế mà TBIR phải đối mặt ngày càng trở nên khó khăn hơn. Nhu cầu này đã hình thành động lực thúc đẩy sự xuất hiện của các kỹ thuật CBIR. Các kỹ thuật và thuật toán được sử dụng có nguồn gốc từ nhiều lĩnh vực như nhận dạng đối tượng và xử lý tín hiệu [17]. Các vấn đề nghiên cứu và phát triển trong CBIR bao gồm nhiều chủ đề, quan trọng nhất là: hiểu nhu cầu của người dùng hình ảnh và hành vi tìm kiếm thông tin, xác định các cách phù hợp để mô tả nội dung hình ảnh, trích xuất các đặc trưng từ hình ảnh thô và khớp truy vấn và hình ảnh được lưu trữ theo cách phản ánh sự tương đồng của con người.

Như minh họa trong **Hình 1.3**, khung CBIR điển hình được chia thành trích xuất đặc trưng ngoại tuyến và tìm kiếm ảnh trực tuyến.

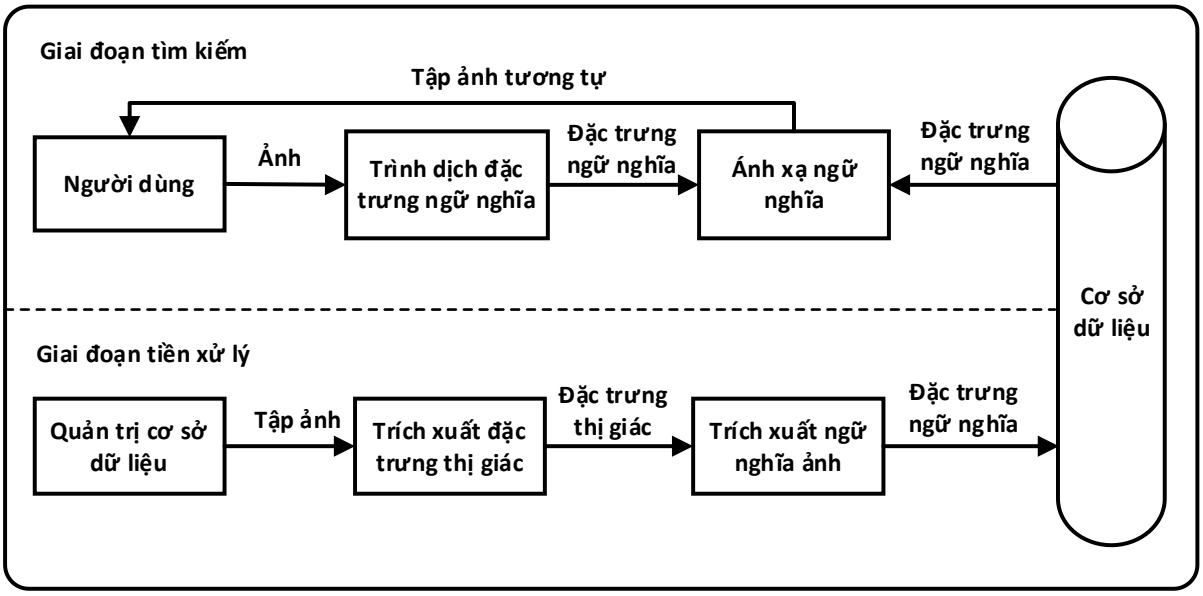


Hình 1.3. Hệ thống tìm kiếm ảnh dựa trên nội dung

❖ ***Tìm kiếm ảnh dựa trên ngữ nghĩa (SBIR)***

Nhìn chung, vấn đề của CBIR là khoảng cách ngữ nghĩa giữa khái niệm ngữ nghĩa cấp cao và hình ảnh cấp thấp như mô tả trong **Hình 1.4**. Nói cách khác, có sự khác biệt giữa những đặc trưng hình ảnh có thể phân biệt và những gì mọi người nhận thức từ hình ảnh. Như mô tả trong **Hình 1.5**, hệ thống SBIR được thực hiện bằng cách trích xuất các đặc trưng cấp thấp của hình ảnh để xác định các vùng hoặc đối tượng có ý nghĩa dựa trên các đặc trưng trực quan. Sau đó, các đặc trưng đối tượng hoặc vùng sẽ được đưa vào quy trình trích xuất ngữ nghĩa hình ảnh để có được mô tả ngữ nghĩa của hình ảnh, sau đó lưu trữ trong cơ sở dữ liệu.

Hình 1.4. Mô tả về "khoảng cách ngữ nghĩa"



Hình 1.5. Hệ thống tìm kiếm ảnh dựa trên ngữ nghĩa

Tìm kiếm ảnh được truy vấn dựa trên khái niệm cấp cao. Truy vấn được thực hiện dựa trên ảnh truy vấn được trích xuất các đặc trưng ngữ nghĩa từ trình dịch các đặc trưng ngữ nghĩa. Quy trình ánh xạ ngữ nghĩa được sử dụng để tìm ra khái niệm tốt nhất để mô tả vùng hoặc đối tượng được phân đoạn dựa trên các đặc trưng cấp thấp. Việc ánh xạ ngữ nghĩa này sẽ được thực hiện thông qua các công cụ học có giám sát hoặc không

giám sát để liên kết các đặc trưng cấp thấp với khái niệm đối tượng và sẽ được chú thích bằng văn bản thông qua quy trình chú thích hình ảnh [18, 19].

1.2. Các đặc trưng phổ biến trong tìm kiếm ảnh

Một đặc trưng được định nghĩa là việc xác định một thuộc tính trực quan của hình ảnh [20]. Nhìn chung, các đặc trưng hình ảnh có thể là toàn cục hoặc cục bộ [21]. Các đặc trưng toàn cục mô tả nội dung trực quan của toàn bộ hình ảnh, trong khi các đặc trưng cục bộ mô tả các vùng hoặc đối tượng (tức là một nhóm nhỏ các pixel) của nội dung hình ảnh. Ưu điểm của việc trích xuất toàn cục là tốc độ cao cho cả việc trích xuất các đặc trưng và tính toán độ tương đồng. Tuy nhiên, các đặc trưng toàn cục thường quá cứng nhắc để biểu diễn một hình ảnh, do đó không xác định được các đặc điểm trực quan quan trọng [20]. Các phương pháp tiếp cận đặc trưng cục bộ hiệu quả truy xuất tốt hơn so với các đặc trưng toàn cục [21]. Chúng biểu diễn hình ảnh có nhiều điểm trong không gian đặc trưng trái ngược với biểu diễn một điểm của đặc trưng toàn cục. Tuy các phương pháp tiếp cận cục bộ cung cấp thông tin mạnh mẽ hơn, nhưng chúng tốn kém hơn về mặt tính toán do không gian đặc trưng của chúng có nhiều chiều và thường sử dụng các thuật toán xấp xỉ như láng giềng gần nhất để thực hiện khớp điểm. Một số đặc trưng quan trọng có thể được sử dụng trong tìm kiếm ảnh như sau:

❖ ***Các đặc trưng màu sắc***

Màu sắc đã được sử dụng rộng rãi trong các hệ thống tìm kiếm ảnh, vì tính toán dễ dàng và nhanh chóng của nó [21]. Màu sắc cũng là một đặc trưng trực quan và đóng vai trò quan trọng trong việc so khớp hình ảnh. Hầu hết các hệ thống tìm kiếm ảnh sử dụng không gian màu, biểu đồ histogram, mô-men, véc-tơ kết hợp màu và mô tả màu chủ đạo biểu diễn màu sắc. Biểu đồ histogram màu là một trong những biểu diễn đặc trưng màu được sử dụng phổ biến nhất trong việc tìm kiếm ảnh. Trong [22] cho rằng việc xác định một đối tượng bằng màu sắc có độ chính xác tốt hơn so với ảnh xám. Mặc dù đặc trưng màu toàn cục dễ tính toán và có thể cung cấp sự phân biệt hợp lý trong việc tìm kiếm ảnh nhưng nó có xu hướng đưa ra quá nhiều kết quả sai khi bộ sưu tập hình ảnh lớn. Nhiều kết quả nghiên cứu cho thấy rằng sử dụng đặc trưng màu cục bộ là giải pháp tốt hơn cho việc tìm kiếm ảnh. Để mở rộng đặc trưng màu toàn cục thành đặc trưng cục bộ,

một cách tiếp cận phổ biến là chia toàn bộ hình ảnh thành các khối con và trích xuất các đặc trưng màu từ mỗi khối con. Ưu điểm của phương pháp này là độ chính xác cao trong khi nhược điểm là vấn đề khó khăn chung của phân đoạn hình ảnh đáng tin cậy [22].

❖ ***Các đặc trưng kết cấu***

Kết cấu là một thuộc tính biểu thị bề mặt và cấu trúc của hình ảnh. Kết cấu có thể được định nghĩa là sự lặp lại đều đặn của một thành phần hoặc mẫu trên bề mặt. Kết cấu hình ảnh là các mẫu hình ảnh phức tạp bao gồm các thực thể hoặc vùng có các đặc điểm về độ sáng, màu sắc, hình dạng, kích thước….Các mô tả kết cấu thường được biết đến là Biến đổi Wavelet, Bộ lọc Gabor và các đặc trưng Tamura [22].

❖ ***Đặc trưng hình dạng***

Hình dạng thường có thể được định nghĩa là mô tả về một vật thể bất kể vị trí, hướng và kích thước của nó. Do đó, các đặc trưng hình dạng phải bất biến với phép tịnh tiến, phép quay và tỷ lệ để có tìm kiếm ảnh hiệu quả. Theo hướng sử dụng hình dạng làm đặc trưng hình ảnh, cần phải xác định ranh giới đối tượng hoặc vùng trong hình ảnh và đây là một thách thức trong xử lý ảnh [23]. So với các đặc trưng màu sắc và kết cấu, các đặc trưng hình dạng thường được mô tả sau khi hình ảnh đã được phân đoạn thành các vùng hoặc đối tượng. Nhìn chung, các biểu diễn hình dạng có thể được chia thành hai loại, dựa trên ranh giới chỉ sử dụng ranh giới bên ngoài của hình dạng và dựa trên vùng sử dụng toàn bộ vùng hình dạng [23]. Các phương pháp đại diện thành công nhất cho hai loại này là các mô tả Fourier và bất biến mô-men.

❖ ***Đặc trưng vị trí không gian***

Vị trí không gian cũng quan trọng và được sử dụng để phân đoạn vùng. Vị trí không gian được mô tả là trên hoặc dưới, trên cùng bên trái hoặc phải và sau hoặc trước theo vị trí của một đối tượng trong hình ảnh. Ví dụ, biển và bầu trời có thể có cùng đặc trưng về kết cấu và màu sắc nhưng thông tin không gian thì không giống nhau. Bầu trời thường biểu diễn phần trên trong khi biển nằm ở phần dưới của hình ảnh. Do đó, thông tin không gian của nhiều đối tượng trong một hình ảnh trích xuất thông tin quan trọng để tìm kiếm ảnh. Hầu hết thông tin không gian được trình bày dưới dạng chuỗi 2D [23].

❖ ***Các đặc trưng hình ảnh cục bộ***

Các đặc trưng cục bộ được phân thành hai loại [23]: (1) các đặc trưng được trích xuất từ ảnh tại các điểm nổi bật và chiều được giảm bằng cách sử dụng phép biến đổi Phân tích thành phần chính (PCA); (2) Các đặc trưng được trích xuất tại các điểm quan tâm. Có ba phương pháp sử dụng các đặc trưng cục bộ cho tìm kiếm ảnh [24]:

- Các đặc trưng cục bộ được trích xuất từ mỗi ảnh cơ sở dữ liệu và từ ảnh truy vấn. Sau đó, các láng giềng gần nhất cho mỗi đặc trưng cục bộ của truy vấn được tìm kiếm và các hình ảnh cơ sở dữ liệu chứa hầu hết các láng giềng này được trả về.
- Các đặc trưng cục bộ từ ảnh truy vấn được so sánh với các đặc trưng cục bộ của mỗi hình ảnh trong cơ sở dữ liệu và khoảng cách giữa chúng được cộng lại. Các hình ảnh có tổng khoảng cách thấp nhất được trả về.
- Các đặc trưng cục bộ từ cơ sở dữ liệu được nhóm lại và sau đó mỗi hình ảnh cơ sở dữ liệu được biểu diễn bằng biểu đồ histogram của các chỉ số của các cụm này. Sau đó, các biểu đồ histogram này được so sánh bằng cách sử dụng phân kỳ Jeffrey.

1.3. Các công trình nghiên cứu liên quan về tìm kiếm ảnh

Phần này cung cấp một bản tóm tắt ngắn gọn về một số công trình liên quan đến TBIR, CBIR và SBIR

1.3.1. Tìm kiếm ảnh dựa trên văn bản

Phương pháp dựa trên văn bản là một phương pháp tìm kiếm đơn giản theo từ khóa truyền thống. Các hình ảnh được lập chỉ mục theo nội dung, như chú thích của hình ảnh; tên tệp, tiêu đề của trang web và thẻ thay thế.... và được lưu trữ trong cơ sở dữ liệu. Một số phương pháp tìm kiếm ảnh dựa trên từ khóa là "túi từ". Một số công cụ tìm kiếm ảnh thương mại, chẳng hạn như Google Image Search và Yahoo Image Search, là các hệ thống tìm kiếm ảnh dựa trên từ khóa. Trong [25] đã đề xuất một phương pháp TBIR có thể khai thác hiệu quả các hình ảnh Web được gắn nhãn để học các bộ phân loại SVM mạnh mẽ. Đầu tiên, phân vùng các hình ảnh Web có liên quan và không liên quan thành các cụm, sau đó xử lý từng cụm các hình ảnh trong mỗi "túi". Để dự đoán nhãn của các

hình ảnh, một lược đồ PMIL-CPB được đề xuất để tự động chọn các túi từ có độ chính xác tốt hơn, dẫn đến các bộ phân loại mạnh mẽ hơn. Nhóm tác giả đã tiến hành các thực nghiệm trên bộ dữ liệu NUS-WIDE và bộ dữ liệu Google, và kết quả chứng minh rõ ràng hiệu quả của phương pháp này.

1.3.2. Tìm kiếm ảnh dựa trên nội dung

❖ ***Đặc trưng toàn cục***

Srivastava và cộng sự [26] đã đề xuất một phương pháp tính toán mềm được gọi là khớp đặc trưng thống nhất (UFM). Trong hệ thống truy xuất này, một hình ảnh được biểu diễn bằng một tập hợp các vùng phân đoạn. Mỗi vùng được đặc trưng bởi một đặc trưng mờ phản ánh các đặc tính màu sắc, kết cấu và hình dạng. Sự giống nhau giữa hai ảnh được định nghĩa là độ tương đồng tổng thể giữa hai họ đặc trưng mờ và được định lượng bằng phép đo độ tương đồng UFM. Phép đo UFM này có hai ưu điểm chính. Ưu điểm đầu tiên là phương pháp tiếp cận UFM làm giảm tác động bất lợi của việc phân đoạn không chính xác. Nó làm cho hệ thống truy xuất mạnh mẽ hơn đối với các thay đổi hình ảnh. Ưu điểm thứ hai là UFM tốt hơn trong việc trích xuất thông tin hữu ích trong cùng các điều kiện không chắc chắn.

Chang và cộng sự [27] đã trình bày một phương pháp tìm kiếm ảnh dựa trên độ tương đồng hình dạng vùng. Phương pháp này bao gồm một số bước. Đầu tiên, các hình ảnh được phân đoạn thành các vùng nguyên thủy. Sau đó, chúng được kết hợp để tạo ra các hình dạng tổng hợp có ý nghĩa, được sử dụng làm đơn vị ngữ nghĩa của các hình ảnh trong quá trình đánh giá độ tương đồng. Công trình sử dụng ba đặc điểm hình dạng toàn cục và một bộ mô tả Fourier được chuẩn hóa để mô tả từng hình dạng có ý nghĩa. Tất cả các đặc điểm này đều bất biến dưới các phép biến đổi tương tự. Cuối cùng, họ đo độ tương đồng giữa hai hình ảnh bằng cách tìm cặp hình dạng giống nhau nhất trong hai hình ảnh. Có hai vấn đề tiềm ẩn với phương pháp này khi xử lý các hình ảnh phức tạp hơn. Vấn đề đầu tiên là máy khó xác định các vùng có ý nghĩa.Vấn đề thứ hai là nhiều đặc trưng và mô hình độ tương đồng được đề xuất, nhưng không có mô hình nào trong số chúng được chứng minh là giống hệt với mô hình thị giác của con người.

Garg và cộng sự [28] đã trình bày một kỹ thuật để trích xuất bản đồ cạnh của một hình ảnh, sau đó là tính toán đặc điểm toàn cục bằng cách sử dụng mức xám cũng như thông tin hình dạng của bản đồ cạnh. Họ sử dụng hình ảnh mờ làm đầu vào và sử dụng khái niệm Trên cùng và Dưới cùng của bề mặt cường độ để trích xuất các ứng cử viên có thể có cho bản đồ cạnh. Độ tương tự giữa các véc-tơ đặc trưng của hai hình ảnh được tính bằng phép đo khoảng cách Euclid.

Deselaers và cộng sự [29] đã thảo luận về nhiều đặc trưng khác nhau để tìm kiếm ảnh và so sánh chúng về mặt định lượng trên bốn nhiệm vụ khác nhau: truy xuất ảnh lưu trữ, truy xuất bộ sưu tập ảnh cá nhân, truy xuất ảnh tòa nhà và tìm kiếm ảnh y tế. Đối với các thực nghiệm của họ, năm cơ sở dữ liệu hình ảnh khác nhau, có sẵn công khai được sử dụng và hiệu suất truy xuất của các đặc điểm được phân tích chi tiết. Điều này cho phép so sánh các đặc trưng phát hiện từ công trình này với các đặc trưng khác chưa được đề cập hoặc sẽ có trong tương lai. Câu hỏi chính được giải quyết trong công trình này là đặc trưng nào phù hợp với nhiệm vụ nào trong tìm kiếm ảnh.

Azimi và cộng sự [30] đã trình bày một phương pháp để khớp đồ thị giống với quá trình suy nghĩ của con người. Hình ảnh được biểu diễn bằng Đồ thị quan hệ thuộc tính mờ (FRAG) mô tả từng đối tượng trong hình ảnh theo tất cả các thuộc tính và mối quan hệ không gian của nó. Họ đề xuất một biểu diễn đặc trưng màu dựa trên các khái niệm mờ. Mô hình đề xuất được áp dụng cho các hình ảnh thực được nhiều người dùng khác nhau đánh giá với các góc nhìn khác nhau và đưa ra kết quả khả quan. Họ thấy rằng vẫn cần phải cải thiện hệ thống được đề xuất này bằng cách sửa đổi các hàm mờ để cải thiện biểu diễn đặc trưng hình ảnh.

Kiamansouri và cộng sự [31] đã tiến hành triển khai và thử nghiệm một thuật toán tìm kiếm và truy xuất dựa trên biểu đồ màu đơn giản cho hình ảnh. Nghiên cứu phát hiện ra rằng kỹ thuật này có hiệu quả khi phân tích bằng phép đo RankPower. Điểm mạnh của thuật toán này là tương đối dễ triển khai theo quan điểm mã hóa. Ngoài ra, hệ thống của họ cho phép tìm kiếm ảnh đã được chuyển đổi về kích thước cũng như được dịch chuyển thông qua phép quay và lật. Nhược điểm chính là việc triển khai biểu đồ màu không nhất thiết cho phép các hình ảnh có liên quan mà thuật toán nhìn thấy giống với các hình

ảnh có liên quan mà con người nhìn thấy. Các kết quả không đồng nhất và không chính xác một cách nhất quán.

Yusof [32] đã trình bày một kỹ thuật mô tả màu chủ đạo (DCD) để tìm kiếm ảnh y tế. Hệ thống hình ảnh y tế thu thập và lưu trữ hình ảnh trong cơ sở dữ liệu y tế. Mục đích của kỹ thuật DCD là tìm kiếm ảnh y tế và hiển thị các hình ảnh tương tự ảnh được truy vấn. DCD chỉ định một số lượng nhỏ các giá trị màu chủ đạo và thống kê. Nó sử dụng khoảng cách Euclidean để khớp với sự tương đồng. Ứng dụng đơn giản đã được phát triển và thử nghiệm bằng cách sử dụng DCD.

❖ ***Các đặc trưng cục bộ***

Alsmadi và cộng sự [33] đã chỉ ra rằng việc khai thác khoảng cách giữa các mô tả cục bộ cải thiện đáng kể độ chính xác của tìm kiếm ảnh. Đầu tiên, họ đã giới thiệu một tiêu chí khoảng cách cung cấp thông tin bổ sung về các kết quả khớp chính xác. Thứ hai, họ khai thác khoảng cách giữa các mô tả SIFT [35] và các láng giềng qua lại để tinh chỉnh thêm phép đo độ tương đồng giữa các mô tả. Phương pháp này vẫn quá tốn kém để áp dụng ngay cả trên vài nghìn hình ảnh.

Xu và cộng sự [34] đã trình bày mô tả hình ảnh cục bộ sử dụng kỹ thuật SIFT lượng tử hóa véc-tơ để tìm kiếm ảnh hiệu quả và hiệu suất hơn. Thay vì biểu đồ hướng có trọng số của SIFT, họ đã áp dụng biểu đồ lượng tử hóa véc-tơ (VQ) làm biểu diễn thay thế cho các đặc trưng SIFT. Họ đề cập rằng các mô tả cục bộ dựa trên VQ mạnh mẽ hơn đối với phép quay, phép biến đổi chiếu và phép chiếu sáng. Kết quả thử nghiệm cho thấy các đặc trưng SIFT sử dụng các mô tả cục bộ dựa trên VQ có thể đạt được độ chính xác tìm kiếm ảnh tốt hơn so với thuật toán thông thường trong khi chi phí tính toán giảm đáng kể.

Winarno và cộng sự [35] đã đề xuất một phương pháp trích xuất đặc điểm cho hệ thống tìm kiếm ảnh dựa trên hệ thống hàm lặp phân vùng tự tương tự cục bộ (PIFS). Đặc trưng của hình ảnh được biểu diễn bằng cách sử dụng khoảng cách và góc của một cặp vị trí khối phạm vi. Hệ thống đề xuất sử dụng độ tương phản biến thể và đặc trưng kích thước không gian để so sánh. Việc thử nghiệm hiệu suất của hệ thống đã sử dụng 1000 hình

ảnh đã được lưu trữ trong cơ sở dữ liệu với 10 danh mục khác nhau với số lượng hình ảnh khác nhau cho mỗi danh mục.

❖ ***Kỹ thuật lai***

Zhang và cộng sự [36] đã đề xuất một hệ thống tìm kiếm ảnh tương tác, tích hợp nội dung văn bản và hình ảnh để nâng cao độ chính xác của truy xuất. Ngoài ra, họ đã trình bày một thuật toán tìm kiếm tinh chỉnh để tối ưu hóa thời gian tìm kiếm của người dùng trên hình ảnh đã truy xuất và cải thiện chất lượng hệ thống. Tinh chỉnh truy vấn bao gồm mở rộng truy vấn và trọng số lại truy vấn. Mở rộng truy vấn cho phép người dùng mở rộng truy vấn để tìm kiếm ảnh mong muốn. Để mở rộng truy vấn, người dùng phải tìm các thuật ngữ có liên quan khác, một lần nữa tốn thời gian và gây mất tập trung trong quá trình tìm kiếm.

An Hồng Sơn và cộng sự [37] đề xuất phương pháp tra cứu ảnh SDAIR với mô hình phân lớp trong CBIR để tăng độ chính xác và giảm thời gian truy vấn. Phương pháp này linh hoạt, không phụ thuộc vào một mô hình học hay độ đo cụ thể, đồng thời có cơ chế tự động bổ sung mẫu dương vào tập huấn luyện mà không cần nhiều mẫu. SDAIR hỗ trợ đồng thời chọn đặc trưng quan trọng và bổ sung mẫu huấn luyện dương. Thực nghiệm trên CSDL ảnh cho thấy phương pháp giúp tăng độ chính xác và tốc độ truy vấn khi dùng RF với cỡ mẫu nhỏ, cỡ lớp nhỏ, và dữ liệu nhiều chiều.

1.3.3. Tìm kiếm ảnh dựa trên ngữ nghĩa

❖ ***Tìm kiếm ảnh theo ngữ nghĩa dựa trên kỹ thuật học máy***

Wu và Hsu [38] đã đề xuất một nền tảng sử dụng thuật ngữ học và các mô tả MPEG-7 để giải quyết các vấn đề phát sinh từ việc biểu diễn và truy xuất ngữ nghĩa của hình ảnh. Nền tảng này cho phép xây dựng nhiều thuật ngữ học gia tăng và chia sẻ thông tin thuật ngữ học thay vì xây dựng một thuật ngữ học duy nhất cho một miền cụ thể không chỉ giữa những người tìm kiếm ảnh mà còn giữa các miền khác nhau. Sự tương đồng giữa thuật ngữ học truy vấn và thuật ngữ học miền để khớp các hình ảnh có liên quan được ước tính bằng cách sử dụng suy luận Naïve Bayesian. Hệ thống này bao gồm ba quy

trình chính: biên dịch RDF, lập chỉ mục truy vấn người dùng và quy trình khớp. Ngoài ra, nó cung cấp một cơ chế phản hồi có liên quan.

Wang và cộng sự [39] đã đề xuất một nền tảng xếp hạng lại hình ảnh mới. Nền tảng này có các phần ngoại tuyến và trực tuyến. Ở giai đoạn ngoại tuyến, các lớp tham chiếu (đại diện cho các khái niệm khác nhau) liên quan đến từ khóa truy vấn được tự động phát hiện và hình được ảnh huấn luyện tự động thu thập trong một số bước. Ở giai đoạn trực tuyến, công cụ tìm kiếm theo từ khóa truy vấn sẽ truy xuất một nhóm hình ảnh. Vì tất cả các hình ảnh trong nhóm đều được liên kết với từ khóa truy vấn theo tệp chỉ mục hình ảnh, nên tất cả chúng đều có chữ ký ngữ nghĩa được tính toán trước trong cùng một không gian ngữ nghĩa do từ khóa truy vấn chỉ định. Họ đã tạo ra ba tập dữ liệu để đánh giá hiệu suất của phương pháp này trong các tình huống khác nhau. Khung đề xuất có thể được cải thiện theo một số hướng: Việc tìm kiếm các phần mở rộng từ khóa được sử dụng để xác định các lớp tham chiếu có thể kết hợp siêu dữ liệu và dữ liệu nhật ký khác ngoài các đặc trưng văn bản và hình ảnh.

Đào Thị Thuý Quỳnh [40] đề xuất hai phương pháp tra cứu ảnh ngữ nghĩa, giải quyết: (1) ảnh kết quả từ nhiều vùng với một truy vấn; (2) không cần phân cụm lại tập phản hồi; (3) xác định độ quan trọng ngữ nghĩa cho truy vấn; (4) tính trọng số đặc trưng; (5) xây dựng hàm khoảng cách từ thông tin địa phương. Phương pháp cải thiện độ chính xác nhưng còn hạn chế khi chưa xét sự không đồng nhất của không gian đặc trưng và không hỗ trợ truy cập xấp xỉ trên không gian non-metric. Thử nghiệm trên tập ảnh COREL với mẫu huấn luyện thu qua RF và ma trận chiếu theo tính địa phương.

W Hu và cộng sự [41] đã đề xuất một một mô hình truy vấn ảnh dựa trên ngữ nghĩa sử dụng phương pháp phân loại hình ảnh dựa trên lựa chọn quan tâm (interest selection). Phương pháp đề xuất tập trung vào việc giải thích véc-tơ riêng của các điểm quan tâm có trọng số và hoàn thành các thử nghiệm nhấp chuột có liên quan để nhận ra việc phân loại các đối tượng cảnh thử nghiệm. Các kết quả thực nghiệm cho thấy lựa chọn quan tâm thứ nhất và thứ hai của đối tượng có tác động lớn đến việc phân loại mục tiêu trong bối cảnh thực nghiệm; phương pháp IWS-SVM có hiệu quả tổng thể tốt nhất đối với việc phân loại đối tượng mục tiêu trong bốn loại cảnh thử nghiệm; phương pháp điểm

quan tâm có thể cải thiện hiệu quả việc truy xuất thông tin hình ảnh. Tuy nhiên, mô hình đề xuất chưa thực hiện phân cụm tập dữ liệu ảnh, dẫn đến việc truy xuất tập ảnh tương tự theo cùng ngữ nghĩa chưa đạt hiệu suất cao. Yupeng Shi và cộng sự [42] đã đề xuất mộ tmô hình tổng hợp dựa trên truy vấn bằng cách tận dụng các hướng dẫn dựa trên truy xuất. Mặc dù phương pháp được đề xuất tổng hợp các hình ảnh thực và vượt trội so với các phương pháp hiện có, nhưng tốc độ suy luận vẫn là một hạn chế và tốc độ truy xuất ảnh tốn nhiều thời gian, khiến nó không thể thực hiện suy luận thời gian thực.

Các phương pháp học máy đã cho thấy tiềm năng lớn trong việc nâng cao độ chính xác và tốc độ của các hệ thống tìm kiếm ảnh, đồng thời cung cấp những giải pháp mới mẻ và hiệu quả cho các thách thức hiện tại. Trong luận án này đã áp dụng một số phương pháp học máy vào việc tìm kiếm ảnh theo mức độ hai nhằm cải thiện hiệu quả của quá trình tìm kiếm dựa trên cấu trúc dữ liệu lưu trữ đề xuất.

❖ ***Tìm kiếm ảnh theo ngữ nghĩa dựa trên ontology***

Thuật ngữ ontology biểu thị cho khoa học về siêu hình học cho phép mô tả bản chất gắn liền với các quan hệ và thuộc tính của nó. Trong khoa học máy tính, ontology đề cập đến tổ chức thông thường của các khái niệm [43]. Ontology có các thuộc tính giải thích cho các khái niệm và các mối quan hệ phi cấu trúc giữa chúng với nhau. Từ vựng về biểu diễn được sử dụng trong ontology. Sự liên kết giữa một thực thể và sự thể hiện của nó là một ký hiệu hoặc một tên mà con người có thể hiểu được bằng trực giác [44]. Ontology bao gồm một phân loại biểu thị một số hỗn hợp của các lớp phổ quát và các mối quan hệ giữa chúng. Chúng ta có thể định nghĩa phân loại như là các thuật ngữ (kiểu hoặc lớp) được liên kết bởi các quan hệ và tất cả chúng được hình thành trong cấu trúc phân cấp.

Fadzli và Setchi [45] đã đề xuất phương pháp tiếp cận ngữ nghĩa để tìm kiếm ảnh dựa trên văn bản cho hình ảnh có chú thích hình ảnh kỹ thuật số theo cách thủ công bằng cách sử dụng các phương pháp thống kê dựa trên DNA ngữ nghĩa (SDNA) được trích xuất từ ontology từ vựng có cấu trúc. Có ba kỹ thuật chính trong cách tiếp cận này: (1) trích xuất SDNA, (2) dựa trên SDNA được trích xuất, phân biệt ý nghĩa từ bằng cách sử dụng các mô hình thống kê, (3) sử dụng SDNA, áp dụng các độ đo tương đồng về ngữ

nghĩa. N. Ruan và cộng sự [46], giới thiệu một nền tảng để tìm kiếm ảnh về ngữ nghĩa từ các kho lưu trữ tập dữ liệu dựa trên ontology phụ thuộc miền. Sử dụng giải thuật không giám sát để trích xuất các đặc trưng màu sắc và kết cấu đạt được vùng đồng nhất được minh họa bằng các khái niệm và được sắp xếp theo ontology dựa trên miền được chỉ định. Đối với các vùng và khái niệm liên quan được sử dụng để triển khai tác vụ truy vấn, kỹ thuật học tương tác sẽ được sử dụng. Manzoor, và cộng sự [43], đã đề xuất một ontology miền cụ thể được liên kết với hệ truy vấn người dùng sử dụng để tìm kiếm ảnh. Người dùng cung cấp cho hệ thống một đầu vào như từ khóa hoặc khái niệm dưới dạng hình ảnh hoặc văn bản. Hệ thống này dựa trên cách tiếp cận kết hợp và sử dụng các cách tiếp cận dựa trên hình dạng, kết cấu và màu sắc được sử dụng cho mục đích phân loại. Sử dụng tập dữ liệu của Mammal để huấn luyện và được thử nghiệm trên miền của Mammal cho việc thử nghiệm. Nhằm nâng cao khả năng tìm kiếm ảnh, Magesh [47] đã áp dụng một khung ngữ nghĩa. Hai cấp độ cần xem xét vấn đề: (1) xác định không gian ngữ nghĩa để tạo ontology, (2) chuyển đổi NLS thành các câu lệnh của ngôn ngữ SPARQL sử dụng truy vấn của nó để truy cập các hình ảnh có liên quan. Biểu mẫu RDF đại diện cho các ontology dựa trên tri thức và tiêu chuẩn dữ liệu hiện có. Liu và cộng sự [48] giới thiệu một mô hình học ngữ nghĩa ontology dựa trên vùng liên kết các loại hình ảnh với các đối tượng trong hình ảnh "CSI". Mỗi mẫu ngữ nghĩa (ST) được xác định trước tương ứng với từng đối tượng, được xác định là đặc trưng màu sắc và kết cấu trung bình của một nhóm các vùng con. Do đó, bằng cách so sánh các đối tượng cục bộ của vùng với tập hợp các ST được xác định trước, các đặc trưng cấp thấp của từng vùng trong ảnh CSI được chuyển đổi thành một đối tượng. Sulaiman và cộng sự [49], trình bày khung hình ảnh ngữ nghĩa ontology đa phương thức, bao gồm bốn thành phần chính: (1) nhận dạng tài nguyên, (2) trích xuất thông tin, (3) xây dựng dựa trên tri thức, và (4) cơ chế tìm kiếm. Các thuộc tính của đối tượng được trích xuất bằng cách tùy chỉnh thuật giải tìm kiếm ảnh ngữ nghĩa. Kết quả thử nghiệm cho thấy kết quả tốt hơn trong cách tiếp cận được đề xuất. Spanier và cộng sự [50] đã xây dựng một ontology đa phương thức MMO (Multi-Modality ontology) nhằm làm giảm khoảng cách ngữ nghĩa hình ảnh bằng cách sử dụng bộ lọc thuộc tính đối tượng OPF (object properties filter). Tuy nhiên, nhóm tác giả chỉ mới xây dựng ontology trên một bộ dữ liệu mẫu nhỏ và thuộc một miền

dữ liệu ảnh cụ thể chưa xây dựng một cấu trúc để lưu trữ dữ liệu hình ảnh.Đồng thời, một thuật toán di truyền được áp dụng kết hợp với tần suất xuất hiện từ trong văn bản để trả về kết quả tìm kiếm. Kết quả thực nghiệm cho thấy mô hình này có hiệu quả trong truy vấn thông tin.Tuy nhiên, mô hình này chưa áp dụng cho tìm kiếm ảnh, cũng như chưa phát triển phương pháp xây dựng ontology tự động hoặc bán tự động để làm giàu dữ liệu. Hơn nữa, tính linh hoạt trong truy vấn chưa được đáp ứng đầy đủ. Mặc dù vậy, mô hình đã đặt nền móng cho việc phát triển một khung ontology xử lý mối quan hệ ngữ nghĩa hình ảnh, với sự tích hợp tự động của HowNet vào ngữ nghĩa dựa trên ontology.

Lư Minh Phúc và Trần Công Án [51] đề xuất một hệ thống tìm kiếm ảnh theo nội dung dựa trên metadata mà còn nâng cao độ phong phú của kết quả nhờ vào sự kết hợp ngữ nghĩa trong tìm kiếm.

Một số hệ thống tìm kiếm ảnh sử dụng ontology cho chú thích, nhưng lại thiếu khả năng sử dụng truy vấn SPARQL hoặc ngôn ngữ tự nhiên để tạo truy vấn SPARQL. Ngoài ra, một số hệ thống khác chỉ xây dựng ontology theo mô hình phân cấp ngữ nghĩa đơn giản mà không giải quyết được việc mở rộng ontology cho bộ dữ liệu hình ảnh, cũng như chưa thể diễn tả đầy đủ ngữ nghĩa theo yêu cầu người dùng.

1.4. Các phương pháp tổ chức thực nghiệm và đánh giá

a. Môi trường thực nghiệm

Các phương pháp và mô hình tìm kiếm ảnh được đề xuất được xây dựng thực nghiệm dựa trên nền tảng dotNet Framework 4.5 và các đường cong đặc trưng ROC được vẽ bằng MatLab 2018. Cấu hình của máy huấn luyện gồm CPU Intel(R) CoreTM i7-9200H 3,5GHz, RAM 32GB và hệ điều hành Windows 10 Professional.

b. Tập dữ liệu ảnh thực nghiệm

Các bộ dữ liệu ảnh tiêu chuẩn được sử dụng trong các thực nghiệm được mô tả trong **Bảng 1.1**.

Bảng 1.1. Các tập dữ liệu ảnh được thực nghiệm trong luận án

STT	Tên tập ảnh	Số lượng ảnh	Số thư mục ảnh	Số lượng lớp ảnh
1	Wang	10.800	80	80
2	ImageCLEF	20.000	39	276
3	MS-COCO	163.957	79	79

Tập ảnh WANG là bộ dữ liệu gồm 10.800 ảnh đơn đối tượng, được phân chia thành 80 chủ đề. Số lượng ảnh trong mỗi chủ đề không đồng đều, từ 100 đến 545 ảnh. Tập ảnh ImageCLEF chứa 20.000 ảnh đa đối tượng, với mỗi ảnh có thể bao gồm nhiều đối tượng thuộc các chủ đề khác nhau. Tập này có 276 lớp, được phân bổ ngẫu nhiên trong 39 thư mục. Để xác định lớp của từng ảnh, các đối tượng trong ảnh được phân vùng, tạo thành tổng cộng 99.535 vùng, trung bình mỗi ảnh có khoảng năm phân lớp đối tượng. Bộ ảnh MS-COCO là bộ ảnh đa đối tượng, gồm 163.957 ảnh chia thành 79 phân lớp. Trong đó, tỷ lệ tập ảnh huấn luyện, kiểm thử và thực nghiệm được trình bày trong **Bảng 1.2**.

Bảng 1.2. Mô tả tỷ lệ tập huấn luyện, kiểm thử và thực nghiệm các bộ dữ liệu

Bộ dữ liệu ảnh	Tổng số lượng ảnh	Số ảnh huấn luyện	Số ảnh kiểm thử	Số ảnh thực nghiệm
Wang	10.800	7.560	2.240	1000
ImageCLEF	20.000	14.000	6.000	1.000
MS-COCO	163.957	114.769	44.188	5.000

c. Các đại lượng đánh giá hiệu suất

Gọi N_δ là tập ảnh liên quan với ảnh tra cứu và có trong tập dữ liệu ảnh, N_ω là tập ảnh đã tìm kiếm được, thì P, R, F_m được tính theo các các công thức (1.1), (1.2) và (1.3)

$$P = \frac{N_\delta \cap N_\omega}{N_\omega} \quad (1.1)$$

$$R = \frac{N_\delta \cap N_\omega}{N_\delta} \quad (1.2)$$

$$F_m = 2 \times \frac{(P \times R)}{(P + R)} \quad (1.3)$$

Xét một truy vấn k, độ chính xác trung bình AP được tính theo công thức (1.4) thường được đo sau khi lấy giá trị trung bình trên các giá trị độ chính xác của từng hình ảnh có liên quan:

$$MAP = \frac{\sum_{k=1}^{N_\delta} P(k) \times R(k)}{N_\delta} \quad (1.4)$$

Trung bình độ chính xác trung bình (MAP) cho một tập các truy vấn N_q bằng giá trị trung bình của các giá trị độ chính xác trung bình cho mỗi truy vấn (q). MAP được tính toán theo công thức (1.5) như sau:

$$MAP = \frac{\sum_{q=1}^{N_q} AP(q)}{N_q} \quad (1.5)$$

Trong đó, $AP(q)$ là độ chính xác trung bình của mỗi truy vấn q và $\mathrm{N_q}$ là số lượng truy vấn thực hiện. Ngoài ra, để mô tả kết quả của hệ tìm kiếm ảnh, đường cong ROC [52, 53] được dùng cho biết tỷ lệ kết quả truy vấn đúng và kết quả truy vấn sai được thực hiện bằng ngôn ngữ Matlab.

1.5. Tiểu kết chương

Chương này cung cấp một cái nhìn tổng quan chi tiết về các phương pháp tìm kiếm ảnh hiện đại, bao gồm hai hướng tiếp cận chính: tìm kiếm ảnh theo nội dung và tìm kiếm ảnh theo ngữ nghĩa. Các phương pháp này không chỉ dựa trên những tiến bộ trong học máy mà còn kết hợp với ontology nhằm mô hình hóa mối quan hệ phức tạp giữa các đối

tượng và khái niệm trong ảnh. Tìm kiếm ảnh theo nội dung tập trung vào việc trích xuất các đặc trưng trực quan từ ảnh như màu sắc, kết cấu, và hình dạng, sau đó sử dụng chúng để so sánh và tìm ra các ảnh tương tự trong cơ sở dữ liệu. Ngược lại, tìm kiếm ảnh theo ngữ nghĩa lại đi sâu vào việc hiểu và suy luận ý nghĩa của các đối tượng trong ảnh, từ đó tạo ra các kết nối giữa các khái niệm và hình ảnh dựa trên bối cảnh ngữ nghĩa của chúng.

Chương này cũng trình bày cụ thể các phương pháp tổ chức thực nghiệm, bao gồm việc thiết lập môi trường thực nghiệm, lựa chọn và chuẩn bị tập dữ liệu thực nghiệm, cũng như các thước đo đánh giá hiệu suất của các phương pháp tìm kiếm. Môi trường thực nghiệm được xây dựng một cách khoa học và tỉ mỉ nhằm đảm bảo tính khách quan và độ tin cậy của kết quả, trong khi các tập dữ liệu thực nghiệm được lựa chọn kỹ càng để phản ánh đúng tính đa dạng và thách thức của các bài toán tìm kiếm ảnh trong thực tế. Các giá trị đánh giá hiệu suất như độ chính xác (precision), độ phủ (recall) và độ đo F (F-measure) được sử dụng để đo lường khả năng của các phương pháp trong việc tìm ra kết quả phù hợp.

Chương tiếp theo sẽ trình bày một cấu trúc dữ liệu mới, đó là cây phân cụm phân cấp, được đề xuất nhằm tối ưu hóa việc lưu trữ và lập chỉ mục tập dữ liệu ảnh. Cấu trúc này không chỉ hỗ trợ quá trình truy xuất nhanh chóng mà còn giúp tăng cường hiệu quả tìm kiếm ảnh theo ngữ nghĩa, cho phép hệ thống có thể xử lý những tập dữ liệu ảnh lớn nhanh và hiệu quả.

CHƯƠNG 2. CẤU TRÚC GP-TREE ĐỂ TÌM KIẾM ẢNH THEO NGỮ NGHĨA

Chương này trình bày các nghiên cứu liên quan đến việc sử dụng cấu trúc cây để lưu trữ và lập chỉ mục cho tập dữ liệu ảnh. Cấu trúc cây phân cụm GP-Tree được mô tả chi tiết, bao gồm các thao tác thêm, sửa, tách, và xóa phần tử. Một mô hình tìm kiếm ảnh theo ngữ nghĩa trên cây GP-Tree dựa trên ontology đã được đề xuất, với mục tiêu cải thiện hiệu suất và độ chính xác của việc tìm kiếm. Hệ thống tìm kiếm ảnh này đã được thử nghiệm trên các bộ dữ liệu phổ biến như Wang, MS-COCO và ImageCLEF, nhằm đánh giá kết quả và tính hiệu quả của mô hình đề xuất. Nội dung của chương này có liên quan trực tiếp đến hai công trình đã công bố là [CT4] và [CT5]; đồng thời cũng liên quan gián tiếp đến các công trình [CT1], [CT2], [CT3].

2.1. Giới thiệu

Lĩnh vực hiện đại của CBIR được chia thành hai loại chính: truy xuất đối tượng cụ thể và truy xuất ở cấp độ danh mục [54]. Trong các hệ thống CBIR gần đây, hình ảnh thường được biểu diễn dưới dạng véc-tơ đặc trưng toàn cục và các khoảng cách giữa các véc-tơ như khoảng cách Euclid được sử dụng để xác định mức độ tương đồng giữa ảnh truy vấn và cơ sở dữ liệu. Nhiều nghiên cứu chuyên sâu về CBIR đã được thực hiện dựa trên các mạng backbone sử dụng các mô hình học sâu khác nhau như mạng nơ-ron tích chập (CNN) [55], bộ chuyển đổi thị giác (ViT) [56] và mạng tích chập đồ thị (GCN) [57] , đóng vai trò như bộ mã hóa để biểu diễn véc-tơ hình ảnh.

Nhu cầu tìm kiếm ảnh dựa trên nội dung hiệu quả đã thúc đẩy nhiều nhà nghiên cứu phát triển các hệ thống CBIR. Các hệ thống đã phát triển này gặp phải nhiều hạn chế và nhược điểm. Đầu tiên là thời gian truy xuất và chi phí tính toán trong nhiệm vụ tìm kiếm ảnh. Nếu độ tương đồng giữa các mô tả của cơ sở dữ liệu và truy vấn được tính toán mà không sử dụng bất kỳ công nghệ lập chỉ mục nào thì thời gian và tài nguyên tính toán cần thiết trở nên rất lớn, đặc biệt khi xử lý các tập dữ liệu có quy mô lớn như trên Internet với số lượng mẫu hình ảnh được lưu trữ đang tăng lên theo cấp số nhân. Do đó, việc áp dụng một hệ thống phân loại cơ sở dữ liệu hoặc cơ chế lập chỉ mục phù hợp rất cần thiết

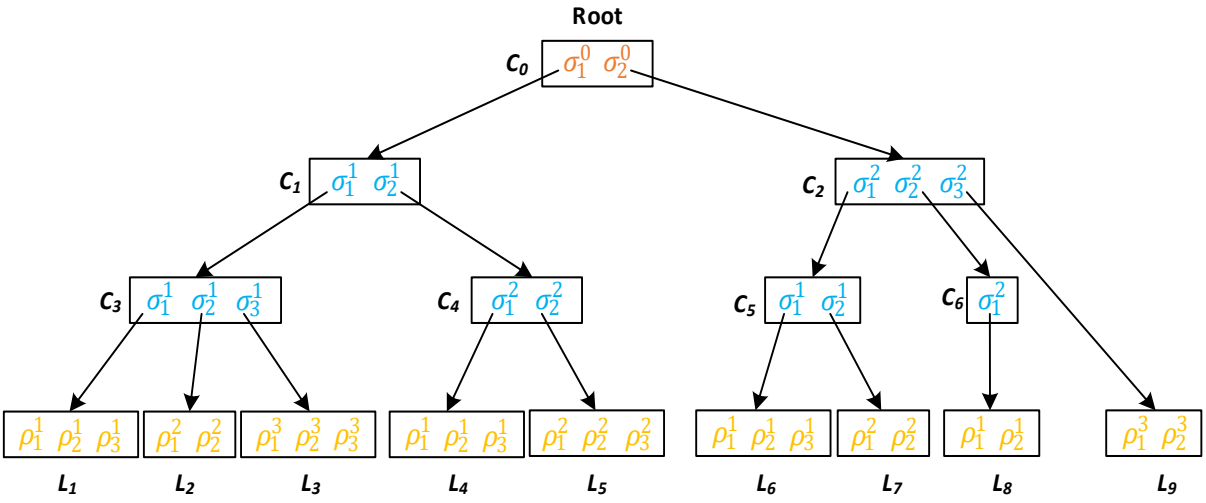
để ứng dụng các hệ thống tìm kiếm ảnh trong các tình huống thực tế. Đặc biệt trong những ứng dụng yêu cầu tìm kiếm ảnh theo thời gian thực, như định vị thị giác (visual localization) với các cảm biến camera được sử dụng để ước lượng vị trí, nên việc giảm thiểu thời gian truy xuất trở thành yếu tố quan trọng. Nghiên cứu tập trung vào việc nâng cao độ chính xác tìm kiếm ảnh là một lĩnh vực phát triển mạnh mẽ và đa dạng [58]. Ngược lại, các nghiên cứu nhằm cải thiện tốc độ tìm kiếm lại ít hơn, với các nghiên cứu gần đây chủ yếu nghiêng về các kỹ thuật dựa trên hashing [59]. Ngoài ra, thách thức của bất kỳ thuật toán CBIR nào là khoảng cách ngữ nghĩa do CBIR chỉ dựa vào đặc trưng thị giác như màu sắc, kết cấu, hình dạng để phân tích và so sánh ảnh, mà không nắm bắt được ý nghĩa ngữ cảnh hoặc mối quan hệ ngữ nghĩa giữa các đối tượng trong ảnh. Điều này gây ra sự không tương thích giữa ý định tìm kiếm của người dùng và kết quả trả về. Chẳng hạn, một người tìm kiếm ảnh "kỳ nghỉ" có thể mong muốn thấy ảnh bãi biển hoặc khách sạn, nhưng CBIR có thể trả về các ảnh màu xanh dương mà không có ngữ nghĩa liên quan.

Việc xây dựng và tìm kiếm trên cây thành hai giai đoạn, bao gồm bước tiền xử lý và tìm kiếm ảnh. Ở bước tiền xử lý, các đặc trưng của cơ sở dữ liệu ảnh được trích xuất và phân cụm chúng theo hệ thống phân cấp. Bước tìm kiếm ảnh bao gồm việc xác định các nút lá tương tự trong cây phân cụm đã được tạo và trích xuất các hình ảnh mục tiêu bằng cách tính toán và xếp hạng mức độ tương đồng giữa truy vấn và cơ sở dữ liệu. Sau khi cấu trúc GP-Tree được xây dựng, khung ontology được tích hợp nhằm bổ sung các yếu tố ngữ nghĩa cho cây phân cấp, từ đó nâng cấp hệ thống từ CBIR sang SBIR. Ontology cung cấp một hệ thống các khái niệm và mối quan hệ giữa chúng, giúp hệ thống hiểu sâu hơn về ngữ cảnh và ý nghĩa của các đối tượng trong hình ảnh, tạo nên một cơ sở ngữ nghĩa để mở rộng phạm vi và mức độ chính xác của các truy vấn. Sự kết hợp giữa GP-Tree và ontology trong SBIR mang lại hai lợi ích rõ rệt: (1) duy trì hiệu suất truy vấn nhờ cấu trúc phân cụm phân cấp của GP-Tree; (2) tăng cường khả năng hiểu và xử lý ngữ nghĩa của hệ thống. Nhờ đó, hệ thống SBIR không chỉ truy xuất nhanh mà còn chính xác hơn, vượt qua những hạn chế về tốc độ và khoảng cách ngữ nghĩa của CBIR truyền thống.

Chương này bao gồm các nội dung chính như sau: Mục 2.2 giới thiệu về cấu trúc dữ liệu GP-Tree; các nguyên tắc thực hiện thao tác trên GP-Tree được trình bày trong Mục 2.3; quy trình xây dựng GP-Tree được mô tả chi tiết ở Mục 2.4; trong Mục 2.5, hệ thống tìm kiếm ảnh dựa trên GP-Tree và phần đánh giá kết quả thực nghiệm sẽ được trình bày. Cuối chương, phần tiểu kết được nêu trong Mục 2.6.

2.2. Cấu trúc dữ liệu GP-Tree

Dựa trên cấu trúc cây từ vựng và phương pháp phân cụm K-Means, cây GP-Tree được xây dựng bằng cách chia tách nút lá thành hai nút nếu số lượng phần tử tại nút đó vượt quá giá trị M định trước. Đồng thời, tại mỗi nút con, có thể áp dụng một ngưỡng θ để đánh giá độ tương tự giữa các phần tử; cụ thể, nếu sai biệt giữa các phần tử vượt quá ngưỡng θ, chúng sẽ thuộc về các nhánh khác nhau. Nhờ đó, GP-Tree phát triển thành một cây đa nhánh, nơi mỗi nút lá là một cụm dữ liệu với các phần tử tương tự nhau. Các phần tử này là các vectơ đặc trưng cho từng ảnh, được lưu trữ trong cây GP-Tree để hỗ trợ thực hiện các thao tác tìm kiếm. **Hình 2.1** mô tả cây phân cụm phân cấp GP-Tree gồm 3 mức.



Hình 2.1. Cây phân cụm phân cấp GP-Tree gồm 3 mức

GP-Tree là một cây đa nhánh gồm nút gốc, nút trong và nút lá, với mỗi nút lá chứa một nhóm ảnh tương tự. Quá trình xây dựng cây GP-Tree bao gồm các thao tác tách nút, thêm phần tử và tạo nhánh mới. Trong quá trình tìm kiếm, cây được duyệt từ nút gốc,

chọn nhánh có phần tử đại diện gần nhất với ảnh truy vấn. Nếu chưa đến nút lá, tìm kiếm tiếp tục ở nút con; khi đến nút lá, cụm ảnh tương tự với ảnh truy vấn sẽ được xác định.

Định nghĩa 2.1: Phần tử dữ liệu ρ tại nút lá là một cặp (τ, f), ký hiệu là $\rho = (f, \tau)$, trong đó $f = (f_1, f_2, \dots, f_n), f_i \in [0,1], \forall i = \overline{1,n}$ là véc-tơ đặc trưng của hình ảnh; τ là đường dẫn chỉ đến tập tin lưu trữ trên trên đĩa (URL).

Định nghĩa 2.2: Phần tử đại diện σ trong nút trong (kể cả nút gốc) là cặp (c, l), ký hiệu là $\sigma = (c, l)$. Trong đó,

- $l = (l_1, l_2, \dots, l_k)$ *gồm k liên kết đến k nút mà nút trong này liên kết*
- $c = (c_1, c_2, \dots, c_n)$ *là giá trị tâm tương ứng với n đặc trưng, mà mỗi* c_i *là trung bình cộng của k giá trị tâm của k nút liên kết.*

Định nghĩa 2.3: Cây GP-Tree là cây bao gồm:

- *Một nút gốc là một tập* C_0 *có* n_0 *phần tử đại diện* $C_0 = \{\sigma_i^0 = (c_i^0, l_i^0) / \forall i = \overline{1, n_0}\}$
- *Một tập T gồm* N_T *nút trong, mà mỗi nút trong là một tập* C_k *có* n_k *phần tử đại diện, ký hiệu* $T = \{C_k = \{\sigma_i^k = (c_i^k, l_i^k) / \forall i = \overline{1, n_k}\} / \forall k = \overline{1, N_T}\}$;
- *Một tập L có* N_L *nút lá, mà mỗi nút lá là một tập* L_l *có* m_l *phần tử dữ liệu, ký hiệu* $L = \{ L_l = \{\rho_i^l = (f_i^l, \tau_i^l) / \forall i = \overline{1, m_l}\} / \forall l = \overline{1, N_L}\}$

Nhận xét: Số hình ảnh được lưu trữ trong một cây GP-Tree được tính bằng tổng số phần tử dữ liệu có trong tất cả các nút lá, được biểu diễn bằng công thức:

$\sum_{l=1}^{N_L} n_l$

Trong đó: N_L: Số lượng nút lá trong cây GP-Tree. Đây là các nút cuối cùng trong cây, mỗi nút lá lưu trữ một số lượng hình ảnh cụ thể; n_l: Số phần tử dữ liệu (hình ảnh) trong nút lá thứ l. Mỗi nút lá chứa một tập các hình ảnh, và n_l đại diện cho số lượng hình ảnh trong nút lá L_l.

Định lý 2.1: Tồn tại duy nhất một nhánh từ gốc đến lá để đưa phần tử ρ vào nút lá mà có các nút trong C_k được xác định bởi.

$$min\left\{\left\|\rho, \sigma_i^k\right\|_2 | \forall i = \overline{1, n_k}\right\}$$

C h ứ n g m i n h:

Gọi η là một nút bất kỳ trên cây GP-Tree, ρ là một phần tử cần thêm vào cây GP-Tree. Nếu $\eta \in T$, theo **Định lý 2.1** thì phần tử ρ luôn chọn được một nhánh con để tạo đường đi đến nút lá. Nếu nhánh con là nút trong thì tiếp tục tìm nhánh con kế tiếp, còn nếu $\eta \in L$ thì đã tìm được nút lá L_l chứa phần tử ρ. Như vậy, luôn tồn tại một nút lá chứa phần tử ρ

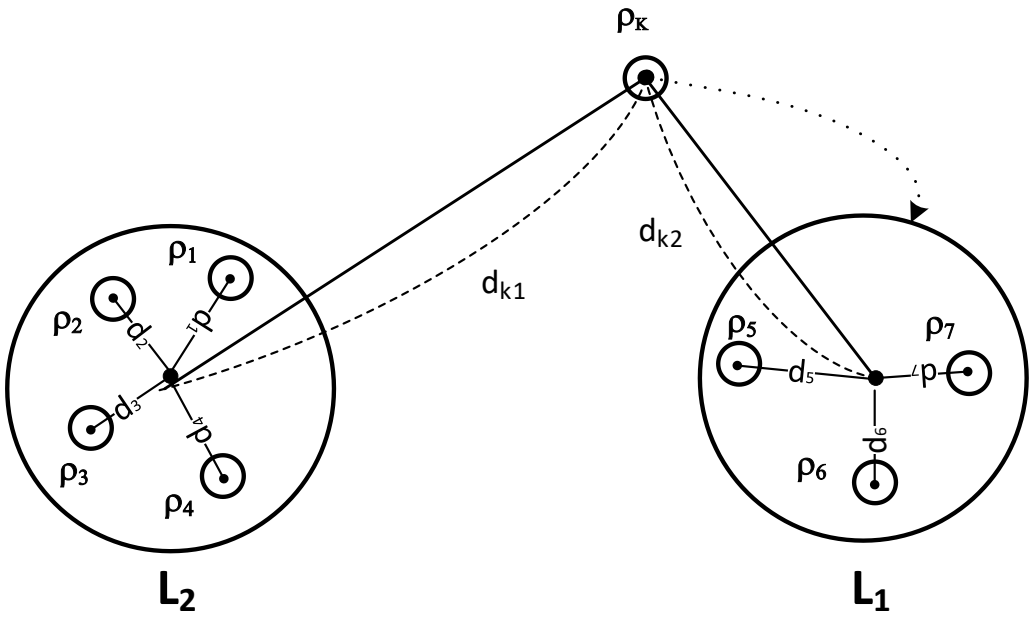
Theo **Định nghĩa 2.3** với mọi nút lá $L_l \in L$ thì luôn tồn tại C_k để nút lá này liên kết đến l^k. Giả sử, có hai nút lá khác nhau $L_u, L_v \in L$ sao cho $\rho \in L_u \wedge \rho \in L_v$. Tức là, có hai đường đi từ C_k để tìm nút lá lưu trữ phần tử ρ. Theo **Định lý 1** thì chỉ chọn được một nhánh đi đến nút con kế tiếp. Vì vậy điều giả sử trên vô lý. Suy ra $L_u \equiv L_v$. Do đó, chỉ có duy nhất một nút lá trên cây lưu trữ phần tử ρ ■

2.3. Các nguyên tắc thực hiện thao tác trên cây GP-Tree

Dữ liệu hình ảnh ngày càng tăng, nên GP-Tree cần có khả năng mở rộng linh hoạt để đáp ứng. Cây phải đảm bảo khả năng lưu trữ phù hợp với sự gia tăng số lượng ảnh, sắp xếp dữ liệu hợp lý và hỗ trợ tìm kiếm nhanh chóng các ảnh tương tự. Các nguyên tắc cho các thao tác thêm, xóa, và tách nút trên GP-Tree được đề xuất như sau:

2.3.1. Thao tác 1: Thêm phần tử dữ liệu vào cây

Ban đầu, nút lá L_1 được khởi tạo rỗng, và các phần tử dữ liệu ρ được thêm vào L_1: $L_1 = \{\rho_i | i = 1..m\}$. Khi số phần tử vượt quá M, nút L_1 sẽ tách, tạo ra một nút gốc mới $C_0 = \left\{\sigma_j^0 | j = 2..n\right\}$. Nút gốc này trở thành nút trong, chứa ít nhất hai phần tử σ. Quá trình tạo cây GP-Tree diễn ra qua việc tách nút và tạo nhánh từ nút lá. **Hình 2.2** minh họa quá trình thêm phần tử mới vào cụm trong nút lá.



Hình 2.2. Ví dụ mô tả thêm phần tử vô cây GP-Tree

Gọi ρ là phần tử dữ liệu cần thêm, η là nút hiện tại trong cây GP-Tree, θ là ngưỡng khoảng cách để xác định nếu ρ có thể được thêm vào nút hiện tại, GP-Tree là cây phân cụm phân cấp hiện tại. Các bước thực hiện thêm một phần tử dữ liệu vào cây GP-Tree như sau:

(1) Kiểm tra nút lá: nếu nút hiện tại η là một nút lá, phần tử ρ sẽ được thêm trực tiếp vào nút lá này.

(2) Kiểm tra số phần tử: sau khi thêm phần tử, nếu số lượng phần tử trong nút lá η vượt quá giới hạn M, thực hiện tách nút sử dụng thuật toán tách nút lá.

(3) Tìm nút con gần nhất: Nếu η không phải là nút lá, thuật toán sẽ tìm nút con C_k có tâm cụm gần với phần tử ρ nhất dựa trên khoảng cách Euclid.

(4) Kiểm tra khoảng cách với ngưỡng θ: Nếu khoảng cách giữa ρ và tâm cụm C_k nhỏ hơn hoặc bằng θ, thuật toán tiếp tục đệ quy và thêm phần tử vào nút lá có nút trong là C_k. Nếu không, khởi tạo một nút lá mới và thêm ρ vào nút lá này.

(5) Cập nhật cây: cập nhật "nút trong" hiện tại và thêm nút lá mới vào cây.

Dựa trên **Định lý 2.1**, thuật toán thêm một phần tử vào cây GP-Tree được mô tả như sau:

Thuật toán 2.1: Thêm phần tử dữ liệu

Input: ρ, η, θ, GP-Tree
Output: Cây GP-Tree sau khi thêm phần tử của nút lá
Function: insertED(ρ, η, θ, GP-Tree)
Begin
  If η is leaf **Then**
    $\eta = \{\eta\} \cup \rho$
    If $|\eta| > M$ **Then**
      // Nếu số phần tử trong nút lá vượt quá giới hạn M, tách nút lá
      GP-Tree = splitLeafNode(η, GP-Tree)
    EndIf
    Return GP-Tree
  EndIf
  // Tìm nút con có khoảng cách Euclid nhỏ nhất đến phần tử ρ
  C_k = argmin(ρ, σ_i^k)
  If ($\|\rho, C_k\|_2 \leq \theta$)
  // Nếu khoảng cách nhỏ hơn ngưỡng θ, tiếp tục thêm phần tử vào cây con gần nhất
    insertED(ρ, C_k, θ, GP-Tree)
  Else
    // Ngược lại, khởi tạo một nút lá mới và thêm phần tử vào đó
    $L_l \leftarrow$ khởi tạo một nút lá mới
    $L_l = L_l \cup \rho$
    $C_k = C_k \cup (\rho, L_l)$
    GP-Tree = GP-Tree $\cup L_l$
  EndIf
End

Tính chất 2.1: Độ phức tạp của **Thuật toán 2.1** là $O(n^2)$.

Chứng minh:

Độ phức tạp thời gian chính của **Thuật toán 2.1** gồm ba bước chính:

(1) Tìm nút con gần nhất: khoảng cách Euclid từ phần tử mới ρ đến tất cả các tâm σ_i^k của các nút con được xác định. Với k nút con, phép tính này mất $O(k)$ thời gian, vì cần tính khoảng cách cho từng tâm và chọn tâm gần nhất.

(2) Tách nút: khi thêm phần tử mới vào nút lá, nếu số lượng phần tử trong nút lá vượt quá ngưỡng M thì nút lá sẽ được phân tách. Quá trình này bao gồm: phân chia các phần tử trong nút lá thành hai nút mới, cập nhật các tâm của các nút mới và cập nhật các nút cha. Thời gian phân tách và cập nhật này là $O(M)$.

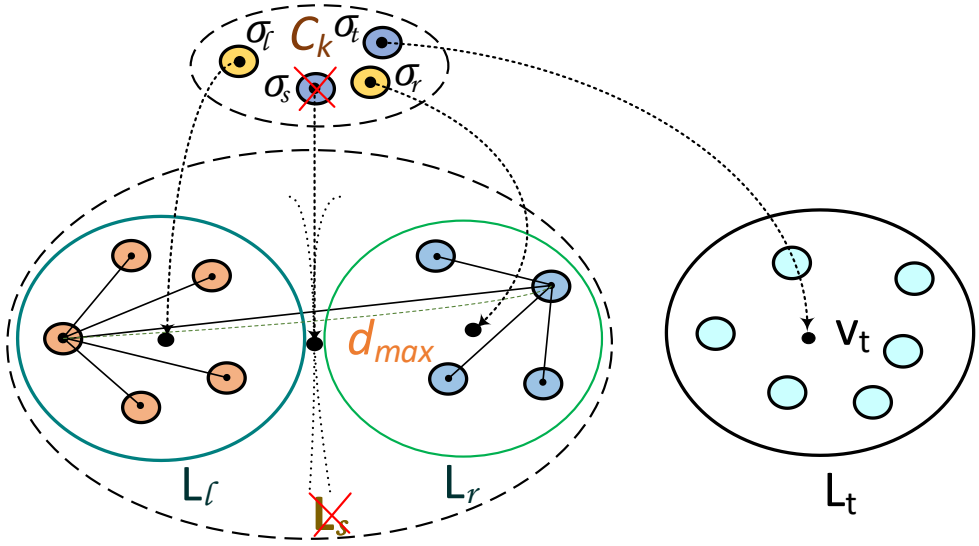
(3) Đệ quy: việc đệ quy đi từ nút cha đến nút con để tìm nút lá thích hợp cho việc thêm phần tử mới. Độ sâu của cây trong trường hợp sẽ là $O(log(n))$, với n là số lượng các nút trong cây. Do đó, thời gian để tìm đúng nút lá để thêm phần tử là $O(log(n))$.

Trong quá trình thêm phần tử vào cây, nếu không cần tách nút, thời gian tìm kiếm nút lá gần nhất mất $O(k)$ với k là số lượng nút con của mỗi nút. Nếu phải tách nút, mỗi lần tách mất $O(M)$ thời gian. Việc đệ quy di chuyển từ gốc cây đến lá mất $O(log(n))$ lần, và mỗi lần thêm phần tử có thể phải tách nút. Do đó, độ phức tạp của **Thuật toán 2.1** có thể là $O(n \times log(n))$ nếu phần lớn các nút không vượt quá ngưỡng. Tuy nhiên, trường hợp xấu nhất là các nút luôn bị tách thì độ phức tạp có thể là $O(n^2)$■.

2.3.2. Thao tác 2: Tách một nút trên cây

Theo **Định nghĩa 2.3**, khi số phần tử tại một nút lá L_l vượt quá số phần tử tối đa M, nút lá sẽ được tách thành hai nút mới. Đồng thời, một nút cha sẽ được tạo ra để liên kết với hai nút lá này, và nút cha này trở thành con của nút cha hiện tại. Các phần tử ρ_i^l sẽ được phân phối giữa hai nút lá mới. Quá trình tách nút lá thành hai nút mới diễn ra qua các bước sau:

- Gọi L_l và L_r là hai nút mới sau khi phân tách nút lá L_s. Xác định tâm nút L_s là giá trị trung bình của các véc-tơ đặc trưng trong L_s.
- Chọn một phần tử ρ_i cách xa tâm của nút L_s làm tâm của nút L_l. Tiếp theo, chọn phần tử ρ_j, phần tử xa ρ_i nhất, là tâm của nút L_r. Sau đó, tạo một nút cha mới (nút trong) của L_l, L_r và thêm hai phần tử tâm ρ_i và ρ_j vào nút cha mới này. (**Hình 2.3**)
- Các phần tử trong nút L_s được phân bổ vào hai nút mới theo quy tắc chọn nút gần nhất dựa trên khoảng cách Euclid. Cập nhật phần tử trung tâm tại cụm nút cha và thực hiện đệ quy tới gốc.



Hình 2.3. Tách nút lá trên cây GP-Tree

Thuật toán tách nút lá trên cây GP-Tree này được mô tả qua **Thuật toán 2.2.**

Thuật toán 2.2: Tách một nút trên cây GP-Tree

Input: L_s, GP-Tree, M
Output: Cây GP-Tree sau khi tách
Function: splitLeafNode(GP-Tree, L_s, M)
Begin
  If số phần tử trong L_s > M **Then**
    // Xác định tâm của nút L_s
    ρ_{center} ← trung bình của tất cả các véc-tơ đặc trưng trong L_s
    // Chọn phần tử cách xa tâm nhất làm tâm cho nút L_l
    ρ_i ← phần tử cách xa nhất với ρ_{center}
    // Chọn phần tử xa ρ_i nhất làm tâm cho nút L_r
    ρ_j ← phần tử xa ρ_i nhất trong L_s
    // Khởi tạo hai nút lá mới L_l và L_r
    L_l ← nút lá mới chứa ρ_i
    L_r ← nút lá mới chứa ρ_j
    // Phân bổ các phần tử còn lại vào L_l và L_rdựa trên khoảng cách Euclid
    Foreach $\rho_k \in L_s$ do
      If $Euclid(\rho_k, \rho_i) < Euclid(\rho_k, \rho_j)$ **Then**
        Thêm ρ_k vào L_l
      Else
        Thêm ρ_k vào L_r
      EndIf

22 **EndForeach**
23 // Tạo nút cha mới và liên kết với L_l và L_r
24 $C_h \leftarrow$ tạo một nút cha mới (nút trong)
25 Liên kết L_l và L_r với C_h
26 Thêm ρ_i và ρ_j vào nút cha C_h
27 // Đệ quy từ nút cha C_h lên gốc để cập nhật tâm nút hiện tại = C_h
28 **While** nút hiện tại có nút cha **do**
29 Cập nhật tâm của nút cha dựa trên các nút con
30 nút hiện tại = nút cha
31 **EndWhile**
32 **Else**
33 **Return** GP-Tree
34 **EndIf**
35 **Return** GP-Tree
36 **End**

Tính chất 2.2: Độ phức tạp của **Thuật toán 2.2** là $O(m_s \times k)$

C h ứ n g m i n h:

Giả sử m_s là số lượng phần tử ρ trong nút lá cần tách L_s, k là kích thước của véc-tơ đặc trưng f của ρ**.** Độ phức tạp thời gian chính của **Thuật toán 2.2** gồm hai bước chính:

(1) Duyệt qua tất cả các phần tử ρ trong L_s: được mô tả từ dòng 16 đến dòng 22 trong thuật toán. Có m_s phần tử trong L_s, mỗi phần tử cần tính toán khoảng cách Euclid với tâm của hai nút mới và so sánh chúng. Độ phức tạp của mỗi phép tính khoảng cách là $O(k)$, nên tổng độ phức tạp cho bước này là $O(m_s \times k)$.

(2) Cập nhật cây GP-Tree với các nút lá mới và nút trong mới: được mô tả từ dòng 28 đến dòng 31 trong thuật toán. Tính các véc-tơ đặc trưng trung bình cho hai nút lá mới L_l và L_r. Tính toán trung bình này có độ phức tạp lần lượt là $O(m_l \times k)$ cho $L_l$và $O(m_r \times k)$ cho L_r, với m_l và m_r lần lượt là số lượng phần tử trong L_l và L_r.

Tổng độ phức tạp của thuật toán là: $O(m_s \times k) + O(m_l \times k) + O(m_r \times k)$. Vì m_s, m_l, và m_r đều phụ thuộc vào số lượng phần tử trong nút lá ban đầu (với $m_l + m_r = m_s$). Do đó, độ phức tạp tổng quát của **Thuật toán 2.2** là $O(m_s \times k)$■

2.3.3. Thao tác 3: Xóa phần tử trên cây

Để xóa một phần tử trên cây, ta cần xem xét phần tử đó là phần tử dữ liệu thuộc nút lá hay phần tử tâm thuộc nút trong.

a. Xóa phần tử thuộc nút lá:

Khi xóa phần tử trong nút lá, thao tác phụ thuộc vào số lượng phần tử còn lại trong nút lá đang xét. Việc xóa có thể làm thay đổi cấu trúc của cây và yêu cầu cập nhật số lượng phần tử và véc-tơ tâm. Hai trường hợp khi xóa phần tử ρ khỏi nút lá L_s:

- **Trường hợp 1:** Khi số lượng phần tử trong nút lá L_s lớn hơn 1 (count(ρ) > 1):
  - Tiến hành xóa phần tử ρ khỏi nút lá L_s.
  - Cập nhật lại số lượng phần tử trong nút lá sau khi xóa.
  - Cập nhật véc-tơ tâm của nút lá và đệ quy lên các nút cha để điều chỉnh véc-tơ tâm từ lá đến gốc, đảm bảo tính nhất quán của cây.
- **Trường hợp 2:** Khi số lượng phần tử trong nút lá L_s chỉ có 1 (count(ρ) = 1):
  - Gán giá trị véc-tơ đặc trưng f của phần tử ρ là $null$, tức là không còn phần tử nào trong nút lá.
  - Cập nhật tâm của nút lá nhưng không bao gồm véc-tơ f trong kết quả tìm kiếm, do nút này đã trống.
  - Nếu sau này có phần tử mới được thêm vào nút lá L_s và số lượng phần tử tăng lên (count(ρ) > 1), tiến hành xóa giá trị $f = null$ để cập nhật lại trạng thái của nút.

Gọi L_s là nút lá chứa phần tử cần xóa và ρ là phần tử cần xóa. Thuật toán xóa phần tử thuộc nút lá trên cây GP-Tree được thực hiện như sau:

Thuật toán 2.3. Xóa một phần tử của nút lá trên cây

```
1  Đầu vào: GP-Tree, L_s, ρ
2  Đầu ra: Cây GP-Tree sau khi xóa phần tử của nút lá.
3  Function: deleteLeafElement(GP-Tree, L_s, ρ)
4  Begin
5      If ρ ∈ L_s Then
6          If count(ρ) > 1 Then
7              // Trường hợp ρ có nhiều hơn 1 phần tử
8              Xóa ρ khỏi L_s
9              Cập nhật số lượng phần tử trong L_s
10         Else
11             // Trường hợp count(ρ) = 1
12             ρ.f = null
13             // Cập nhật véc-tơ tâm
14             Cập nhật tâm của L_s (không bao gồm ρ)
15         EndIf
16         // Cập nhật véc-tơ tâm từ lá đến gốc
17         nút hiện tại = L_s
18         While nút hiện tại có nút cha do
19             Cập nhật tâm của nút cha dựa trên các nút con còn lại
20             nút hiện tại = nút cha
21         EndWhile
22     Else
23         Return GP-Tree
24     EndIf
25     Return GP-Tree
26 End
```

Tính chất 2.3: Độ phức tạp của **Thuật toán 2.3** là $O(M \times h)$

Chứng minh:

Giả sử GP-Tree có chiều cao là h, Mỗi nút lá chứa tối đa M phần tử, số lượng phần tử trong nút lá là $count(\rho)$. Độ phức tạp thời gian chính của **Thuật toán 2.3** gồm 4 bước chính:

- Kiểm tra phần tử ρ trong nút lá L_s: việc này mất $O(M)$ nếu cần duyệt qua tất cả phần tử trong nút lá.
- Xóa phần tử ρ khỏi L_s: Nếu $count(\rho) > 1$, việc xóa ρ khỏi nút lá L_s có thể mất O(M) khi cập nhật danh sách phần tử trong nút lá. Nếu $count(\rho) = 1$, việc

gán $\rho.f = null$ sẽ tốn $O(1)$, vì đây chỉ là thao tác gán một giá trị rỗng cho một phần tử.

- Cập nhật véc-tơ tâm của nút lá L_s: việc cập nhật bằng cách tính lại trung bình các véc-tơ đặc trưng trong nút lá, ngoại trừ ρ. Việc tính toán này có độ phức tạp $O(M)$ vì cần duyệt qua các phần tử khác trong nút lá.
- Cập nhật véc-tơ tâm từ lá đến gốc: sau khi xóa phần tử, tâm của các nút cha từ nút lá L_s đến gốc cần được cập nhật. Độ phức tạp của việc này phụ thuộc vào chiều cao của cây h. Mỗi lần cập nhật một nút cha tốn $O(M)$ để tính toán lại tâm của nút đó dựa trên các nút con. Do có tối đa h nút cha, quá trình này mất $O(M \times h)$.

Trong trường hợp thuật toán phải duyệt qua tất cả các phần tử trong nút lá để tìm ρ, tiến hành xóa ρ và cập nhật lại cây từ lá đến gốc. Do đó, độ phức tạp của **Thuật toán 2.3** là $O(M \times h)$■

b. Xóa phần tử thuộc nút trong:

Khi tiến hành xóa nút lá thì phần tử đại diện của nút lá đó cũng phải bị xóa. Gọi GP-Tree là cây GP-Tree hiện tại, C_s là nút trong chứa phần tử σ, σ là phần tử cần xóa và là tâm của một nút lá hoặc nút trong cấp thấp hơn. Các bước thực hiện xóa một nút trong như sau:

- Kiểm tra xem phần tử σ có trong nút trong C_s. Nếu không tồn tại, thông báo và kết thúc thuật toán.
- Xóa phần tử σ khỏi nút trong C_s.
- Xóa nút con tương ứng với đường dẫn l (đường dẫn mà σ đại diện).
- Bắt đầu từ nút hiện tại C_s, tiếp tục cập nhật số lượng phần tử và tâm của các nút cha, đệ quy lên đến gốc cây.
- Cập nhật số lượng phần tử và véc-tơ tâm từ các nút liên quan, đồng thời đệ quy cập nhật tâm của cây từ dưới lên trên (từ nút con đến gốc).

Thuật toán xóa phần tử của một nút trên cây được thực hiện như sau:

Thuật toán 2.4. Xóa một phần tử của nút trong trên cây

```
1   Đầu vào: GP-Tree, C_s, σ
2   Đầu ra: Cây GP-Tree sau khi xóa phần tử của nút lá.
3   Function: deleteInternalNodeElement(GP-Tree, C_s, σ)
4     Begin
5       If σ ∈ C_s Then
6         Xóa σ khỏi C_s
7         // Xóa nút con tương ứng với đường dẫn l
8         l = đường dẫn tương ứng với σ
9         Xóa nút con tại l
10        // Cập nhật số lượng và tâm của nút cha và các nút trong liên quan
11        nút hiện tại = C_s
12        While nút hiện tại có nút cha do
13          Cập nhật số lượng phần tử σ của nút cha
14          Cập nhật tâm của nút cha dựa trên các nút con còn lại
15          nút hiện tại = nút cha
16        EndWhile
17        // Cập nhật nút lá nếu cần thiết
18        If nút con là nút lá Then
19          Cập nhật lại tâm của các nút lá tương ứng
20        EndIf
21      Else
22        // Nếu σ không tồn tại trong nút trong
23        Return GP-Tree
24      EndIf
25      Return GP-Tree
26  End
```

Tính chất 2.4: Độ phức tạp của **Thuật toán 2.4** là $O(N \times h)$.

Chứng minh:

Giả sử GP-Tree có chiều cao là h, số phần tử đại diện trong mỗi nút trong là N. Độ phức tạp thời gian chính của **Thuật toán 2.4** gồm 2 bước chính: (1) Tìm và xóa phần tử σ trong nút trong C_s mất $O(N)$; (2) Cập nhật tâm và số lượng phần tử: được mô tả từ dòng 12 đến dòng 16 trong thuật toán; mỗi lần cập nhật các nút cha mất $O(N)$, và việc này diễn ra trong h tầng của cây. Do đó, độ phức tạp của **Thuật toán 2.4** là $O(N \times h)$■.

2.4. Tạo trúc dữ liệu GP-Tree

Trên cơ sở **Định nghĩa 2.3** và **Định lý 2.1**, cây GP-Tree là cây đa nhánh và tăng trưởng theo hướng lá. Cây GP-Tree được tạo bằng cách thêm từng phần tử dữ liệu vào trong cấu trúc của cây. Phần tử được thêm chỉ chọn duy nhất một hướng trên cây để xác định nút lá để lưu trữ. Do đó, nếu đi từ nút gốc đến nút lá thì chỉ chọn được một nút lá phù hợp nhất để lưu trữ. Quá trình thêm này sẽ thực hiện việc tách nút và cây sẽ tăng trưởng để chứa bộ dữ liệu ban đầu.

Để tạo cây GP-Tree cần sử dụng các thuật toán thêm phần tử vào cây (Thuật toán 2.1), Thuật toán tách nút trên cây (Thuật toán 2.2) và thuật toán xoá một phần tử trên cây (Thuật toán 2.3 và Thuật toán 2.4). Gọi n là số lượng phần tử trong tập Γ, quá trình tạo cây GP-Tree được mô tả qua **Thuật toán 2.5**.

Thuật toán 2.5: Tạo cây GP-Tree

Input: Tập dữ liệu ảnh Γ, ngưỡng θ
Output: GP-Tree
Function: createGPT(Γ, θ)
Begin
    GP-Tree = null
    $L_0 \leftarrow$ khởi tạo mộ nút lá
    Foreach ($\rho \in$ Γ) **do**
        GP-Tree = **insertED**(ρ, L_0, θ,GP-Tree)
    EndForeach
    Return GP-Tree
End

Tính chất 2.5: Độ phức tạp của **Thuật toán 2.5** là $O(n^2 \times M)$.

Chứng minh:

Độ phức tạp của **Thuật toán 2.5** phụ thuộc vào độ phức tạp của hai yếu tố chính là vòng lặp duyệt qua từng phần tử trong tập dữ liệu và hàm $insertED$.

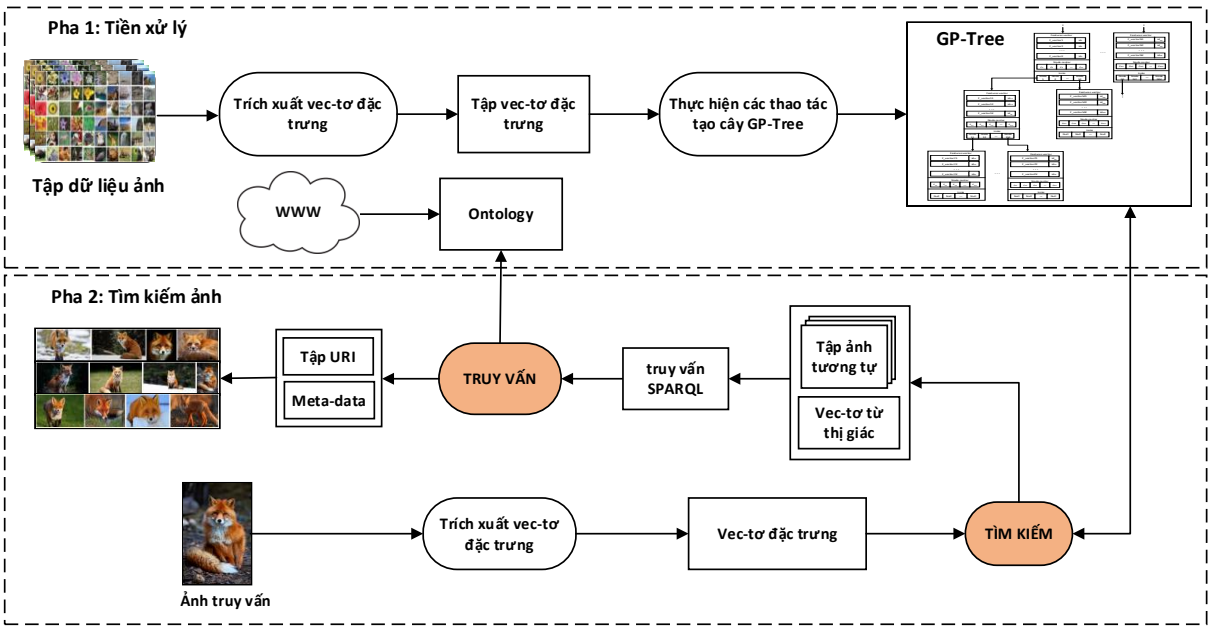
- Vòng lặp duyệt qua tập dữ liệu ảnh: Giả sử tập dữ liệu ảnh Γ chứa n phần tử. Vòng lặp $Foreach(\rho \in \Gamma)$ thực hiện n lần, vì mỗi phần tử trong tập dữ liệu đều phải được thêm vào cây GP-Tree.
- Hàm insertED: Độ phức tạp của hàm insertED là yếu tố chính quyết định độ phức tạp của toàn bộ thuật toán. Nếu tập dữ liệu không phân cụm tốt hoặc có quá nhiều phần tử trong một nút, cây có thể trở thành không cân bằng, dẫn đến độ phức tạp là $O(M \times h)$ cho mỗi lần thêm một phần tử, trong đó M là số phần tử tối đa trong nút lá và h là chiều cao của cây.

Nếu cây GP-Tree phân nhánh tốt và cân bằng, chiều cao của cây h có thể xấp xỉ $log(n)$, trong trường hợp này, độ phức tạp có thể là $O\big(n \times M \times log(n)\big)$. Tuy nhiên, nếu cây không cân bằng và trở nên giống cây đường thẳng, thì h có thể xấp xỉ n, dẫn đến độ phức tạp tệ nhất là $O(n \times M \times n)$, tức là $O(n^2 \times M)$■.

2.5. Tìm kiếm ảnh dựa trên cấu trúc GP-Tree

2.5.1. Hệ tìm kiếm ảnh dựa trên cây GP-Tree

Hệ thống tìm kiếm ảnh ngữ nghĩa sử dụng GP-Tree và ontology, gọi là GP-SBIR, bao gồm hai giai đoạn: (1) Tiền xử lý, tạo cây GP-Tree và xây dựng khung ontology bán tự động dựa trên ngôn ngữ RDF, lưu trữ dưới định dạng N3; (2) Tìm kiếm ảnh tương tự, trích xuất phân lớp từ tập ảnh tương tự để tạo véc-tơ thị giác, sau đó tạo câu truy vấn SPARQL để truy vấn ontology, từ đó lấy các ảnh tương tự và chú giải ảnh truy vấn.



Hình 2.4. Mô hình hệ tìm kiếm ảnh dựa trên GP-Tree (GP-SBIR)

Mô hình tìm kiếm ảnh đề xuất gồm giai đoạn tiền xử lý và giai đoạn tìm kiếm được mô tả như hình **Hình 2.4**. Trong đó, cấu trúc GP-Tree và giai đoạn tìm kiếm ảnh là phần đề xuất; phần ontology là kế thừa từ công trình [60] sau khi bổ sung dữ liệu thực nghiệm bộ ảnh WANG, MS-COCO. Tập từ thị giác được xây dựng trên cơ sở tính theo số ảnh có cùng số phân lớp nhiều nhất trong tập ảnh tương tự theo nội dung đã tìm được từ GP-Tree.

a. Phân đoạn ảnh và xác định phân lớp các đối tượng trong ảnh

Trong luận án này, mô hình Mask R-CNN đã được huấn luyện trước để phát hiện các đối tượng trong ảnh và xác định tập phân lớp cho ảnh đầu vào. **Hình 2.5** minh họa kết quả nhận dạng và phân lớp các đối tượng trên bộ dữ liệu MS COCO bằng Mask R-CNN với ResNet-101-FPN.

Hình 2.5. Kết quả của Mask R-CNN sử dụng ResNet-101-FPN trên các ảnh trong bộ dữ liệu COCO

Kết quả so sánh giữa Mask R-CNN và các phương pháp phân đoạn ảnh hiện đại khác trên bộ dữ liệu MS-COCO Test-dev được trình bày trong [61]. Các mô hình phân đoạn ảnh hiệu quả trên bộ dữ liệu này là MNC và FCIS, tuy nhiên Mask R-CNN vượt trội hơn FCIS khi thử nghiệm với nhiều kích thước ảnh khác nhau.

b. Trích xuất đặc trưng và biểu diễn hình ảnh

Việc chuyển đổi hình ảnh kỹ thuật số trực tiếp thành biểu diễn số tạo ra ma trận chiều cao, không phù hợp cho việc phân loại hình ảnh hoặc lập chỉ mục hình ảnh. Thay vào đó, trong hệ thống CBIR, sử dụng biểu diễn hình ảnh dựa trên các mẫu đã học từ nội dung cấp thấp để lập chỉ mục hình ảnh. Phương pháp này giúp so sánh nhanh giữa các mẫu do người dùng cung cấp và hình ảnh trong tập dữ liệu ảnh để tìm kiếm hiệu quả [62]. **Hình 2.6** minh họa quá trình trích xuất đặc trưng của một ảnh trong bộ dữ liệu MS-COCO.

Hình 2.6. Trích xuất đặc trưng ảnh 000000133819 trong bộ dữ liệu ảnh MS-COCO

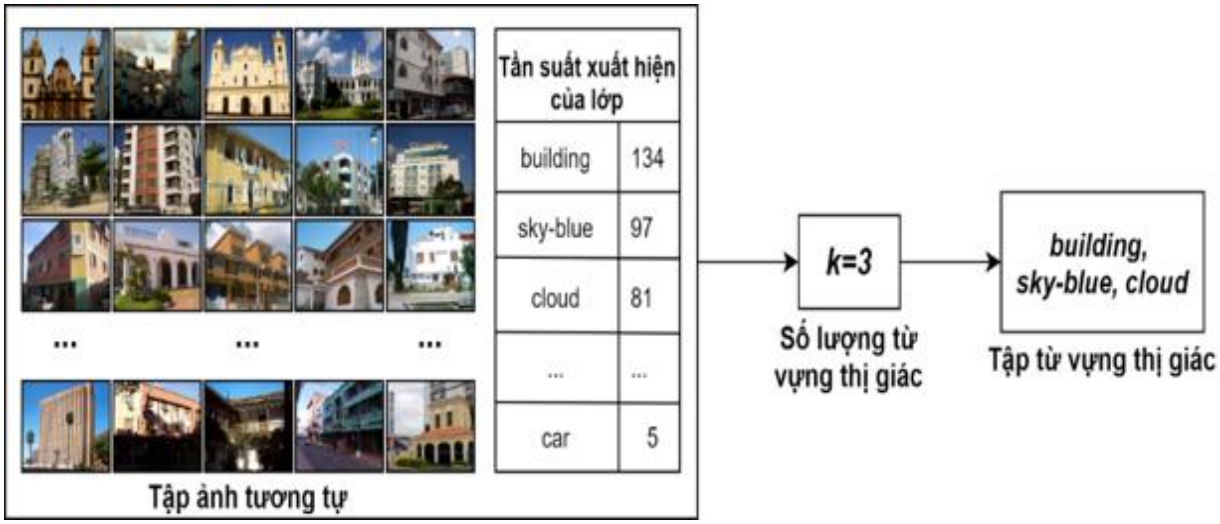
Bảng 2.1 mô tả số thành phần của các đặc trưng trong véc-tơ đặc trưng đại diện cho một ảnh. **Hình 2.6** mô tả trích xuất các đặc trưng của ảnh 000000133819 trong bộ dữ liệu ảnh MS-COCO.

Bảng 2.1. Mô tả số thành phần của các đặc trưng trong véc-tơ đặc trưng đại diện cho một ảnh

Đặc trưng	Số thành phần
Đặc trưng màu theo dải màu Newton.	25
Đặc trưng cường độ sáng của ảnh dựa trên cường độ láng giềng trội.	25
Đặc trưng diện tích của đối tượng và hình nền.	25
Đặc trưng vị trí tương đối của đối tượng.	9
Đặc trưng vị trí tương đối của hình nền.	9
Đặc trưng đường biên đối tượng bằng phép lọc Laplace.	16
Đặc trưng chu vi của đối tượng và hình nền.	16
Đặc trưng bề mặt đối tượng bằng phép lọc Sobel.	16

c. Trích xuất từ vựng thị giác của ảnh

Véc-tơ từ vựng thị giác đại diện cho các lớp ngữ nghĩa phổ biến trong bộ ảnh tương tự, giúp xác định các đối tượng dự đoán trong ảnh. Số lượng từ vựng này phụ thuộc vào số vùng và lớp trong ảnh đầu vào. **Hình 2.7** minh họa ví dụ về từ vựng thị giác trích xuất từ bộ ảnh tương tự.



Hình 2.7. Một ví dụ cho tập từ vựng thị giác

Gọi Ω là tập các hình ảnh tương tự, L là tập các phân lớp trong tập Ω, γ là ngưỡng tần suất mà một lớp, **Thuật toán 2.6** mô tả quá trình tạo tập từ vựng thị giác như sau:

Thuật toán 2.6: Tạo tập từ vựng thị giác

Input: Tập các hình ảnh tương tự Ω, ngưỡng γ
Output: Tập từ vựng thị giác W
Function: $CreateVW(\Omega, \gamma)$
Begin
    Gọi L là tập các lớp trong Ω;
    $W = \emptyset$;
    For L_i là một thuộc tập L **do**
        If tần suất xuất hiện của L_i lớn hơn hoặc bằng ngưỡng γ **Then**
            Thêm L_i vào tập từ vựng thị giác W;
        EndIf
    EndFor
    Return W;
End.

Tính chất 2.6: Độ phức tạp của **Thuật toán 2.6** là $O(n)$.

C h ứ n g m i n h:

Giả sử n là số lượng lớp trong tập L. Với mỗi lớp L_i, thuật toán kiểm tra xem tần suất xuất hiện của lớp đó có lớn hơn hoặc bằng ngưỡng γ hay không. Giả sử việc đếm tần suất xuất hiện của L_i mất thời gian $O(n)$, trong đó n là tổng số hình ảnh trong tập Ω.Thuật toán thực hiện n lần để lấy ra n từ vựng thị giác. Do đó, độ phức tạp của **Thuật toán 2.6** là $O(n)$■.

d. Khung ontology cho bài toán tìm kiếm ảnh theo mức độ hai

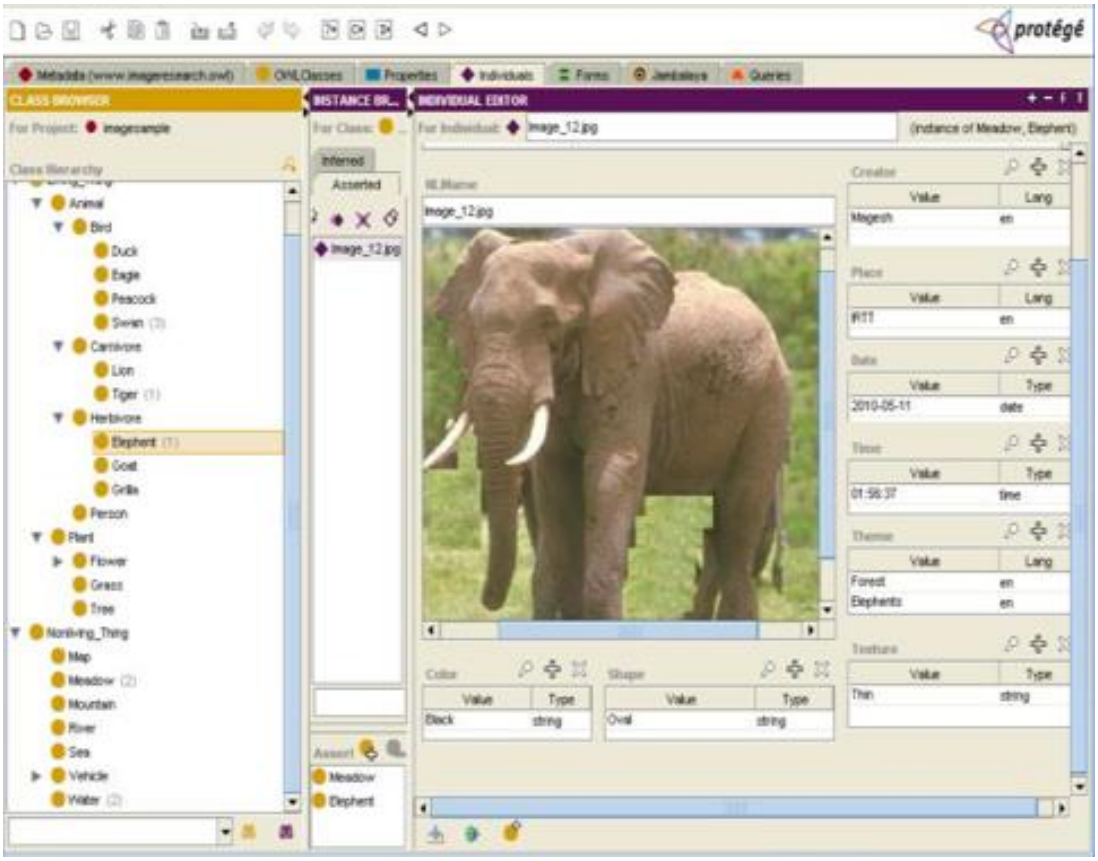
Ontology được phát triển dựa trên tập ảnh đối tượng và mối quan hệ giữa chúng. Các ảnh đa đối tượng được phân đoạn thành ảnh đối tượng, trích xuất các thành phần và xây dựng mối quan hệ giữa các đối tượng. Mô hình ontology được tạo ra và bổ sung các thuộc tính của ảnh đối tượng cùng với các mối quan hệ này. Trong luận án, ảnh đối tượng được phân đoạn bằng mạng R-CNN, với mỗi ảnh có số lượng đối tượng khác nhau. Mạng R-CNN được sử dụng để nhận diện và phân đoạn ảnh thành các ảnh đối tượng, kế thừa mô hình đã có để phân đoạn các đối tượng trên ảnh gốc [63]. Mục tiêu chính của ontology là biểu diễn hình ảnh theo ngữ nghĩa, giúp việc tìm kiếm ảnh trở nên dễ dàng hơn.

❖ ***Mô hình khung ontology***

Các ảnh trong tập dữ liệu được biểu diễn ở cấp độ cao, với mỗi ảnh được chú thích bằng các danh mục mô tả như vị trí hoặc nhãn. **Hình 2.8** mô tả việc gắn ảnh cho các phân lớp trong ontology. Quá trình tìm kiếm bắt đầu từ nút gốc của cây ontology, là *Owl: Thing*, và mỗi lớp được kết nối qua quan hệ kế thừa. Ý nghĩa của hình ảnh được xác định thông qua chỉ mục lớp. Truy vấn được chuyển đổi thành cấu trúc RDF và so sánh với ontology, giúp xác định và truy xuất những ảnh có cùng danh mục.

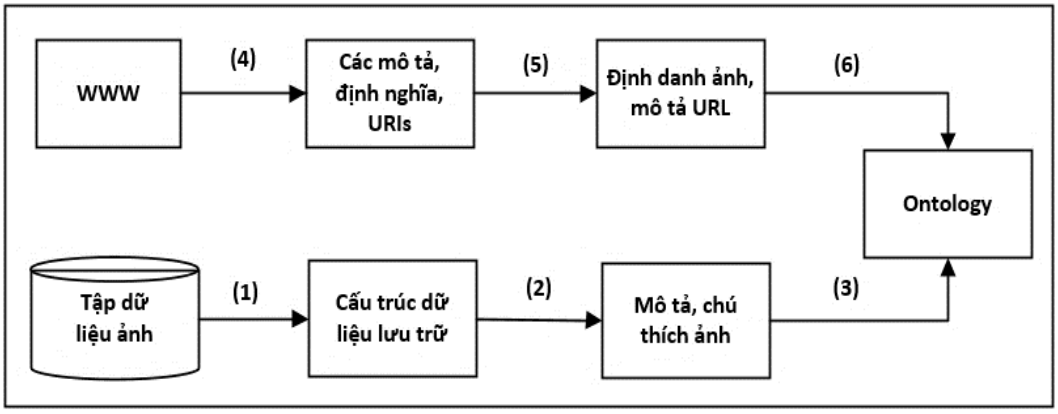
Mô hình xây dựng khung ontology cho dữ liệu ảnh được đề xuất dựa trên bộ ảnh MS-COCO, có sự phân cấp lớp và quan hệ giữa các ảnh và phân lớp. Quá trình xây dựng

bao gồm các bước: (1) Kế thừa các phân lớp từ MS-COCO và bổ sung phân cấp lớp; (2) Thêm chú thích và mô tả phân lớp cho ảnh; (3) Lấy URI và định nghĩa lớp từ WWW và WordNet để bổ sung cho bộ ảnh ImageCLEF; (4) Tạo khung ontology cho ảnh MS-COCO và các thông tin bổ sung từ WWW.



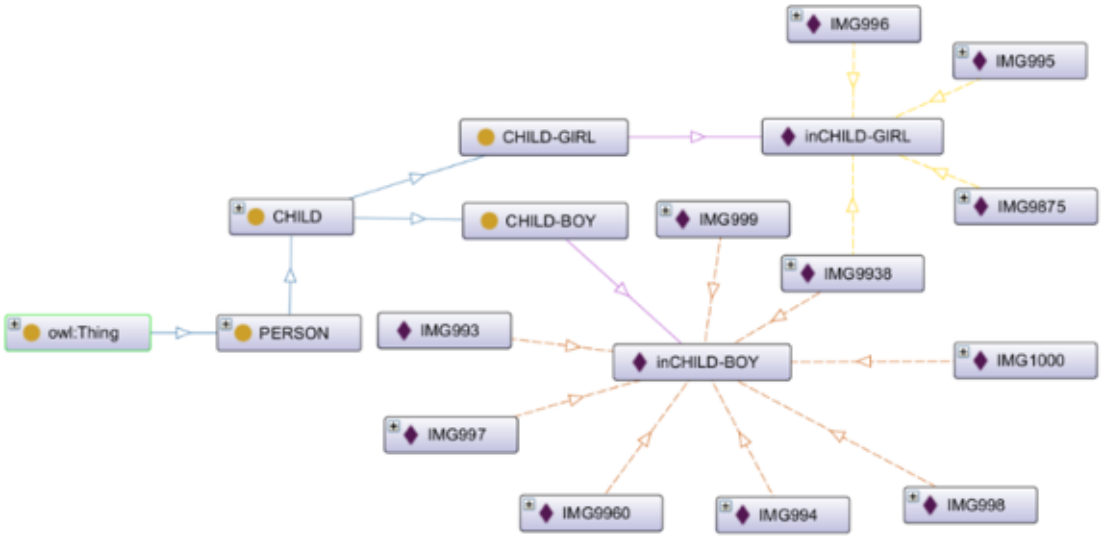
Hình 2.8. Gắn ảnh cho các phân lớp trong ontology

Ontology cho ảnh được xây dựng trên nền tảng Protégé và lưu trữ dưới định dạng OWL hoặc RDF/XML. Mô hình đề xuất, như mô tả trong **Hình 2.9**, giúp xây dựng ontology mẫu. Tuy nhiên, Protégé chỉ hỗ trợ trực quan hóa mà không hiệu quả trong việc quản lý các cơ sở tri thức lớn, như dữ liệu ảnh. Do đó, cần phương pháp tối ưu để xây dựng và quản lý cơ sở dữ liệu ảnh quy mô lớn.



Hình 2.9. Mô hình xây dựng khung ontology

Các phân lớp ảnh được tổ chức theo cấu trúc phân cấp, với từ điển ngữ nghĩa từ WordNet. Mỗi ảnh là thể hiện của một hoặc nhiều phân lớp trong ontology. **Hình 2.10** minh họa một ví dụ về ontology thiết kế cho bộ dữ liệu MS-COCO.

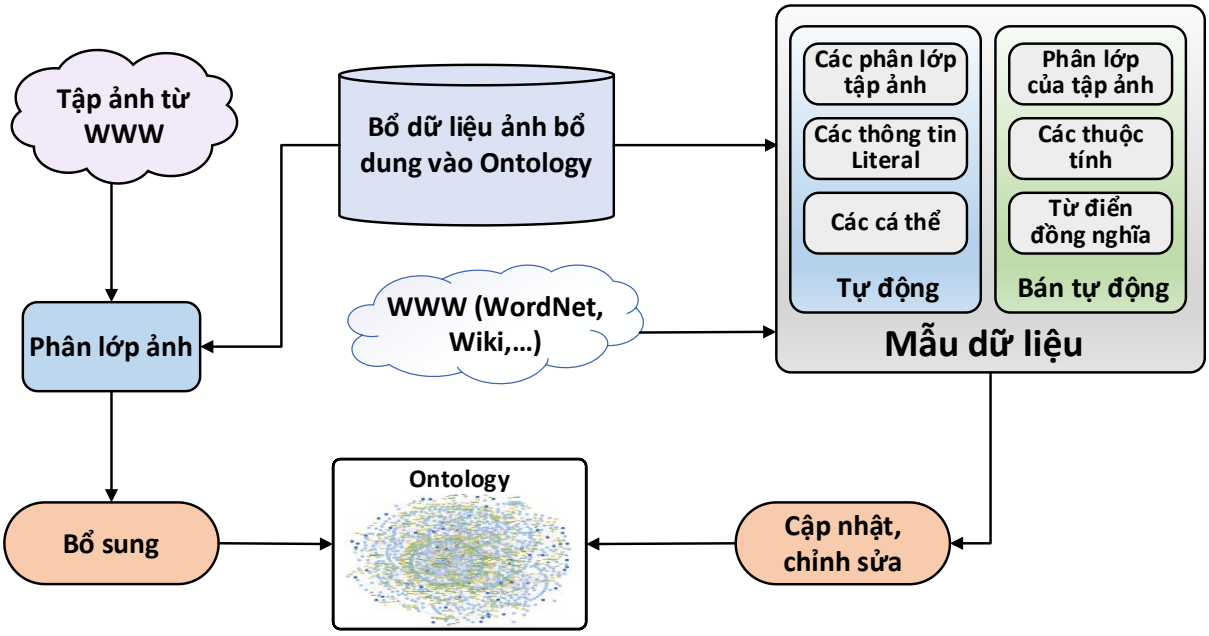


Hình 2.10. Một ví dụ về ontology áp dụng trên bộ dữ liệu ảnh MS-COCO

Phương pháp thủ công để xây dựng ontology bằng công cụ như Protégé mang lại độ chính xác cao nhưng tốn nhiều thời gian và công sức, không phù hợp với cơ sở dữ liệu ảnh lớn. Phương pháp tự động, mặc dù nhanh chóng và không cần can thiệp, nhưng dữ liệu ảnh từ Internet thường phân tán và thiếu đồng nhất, làm giảm độ tin cậy của ontology.

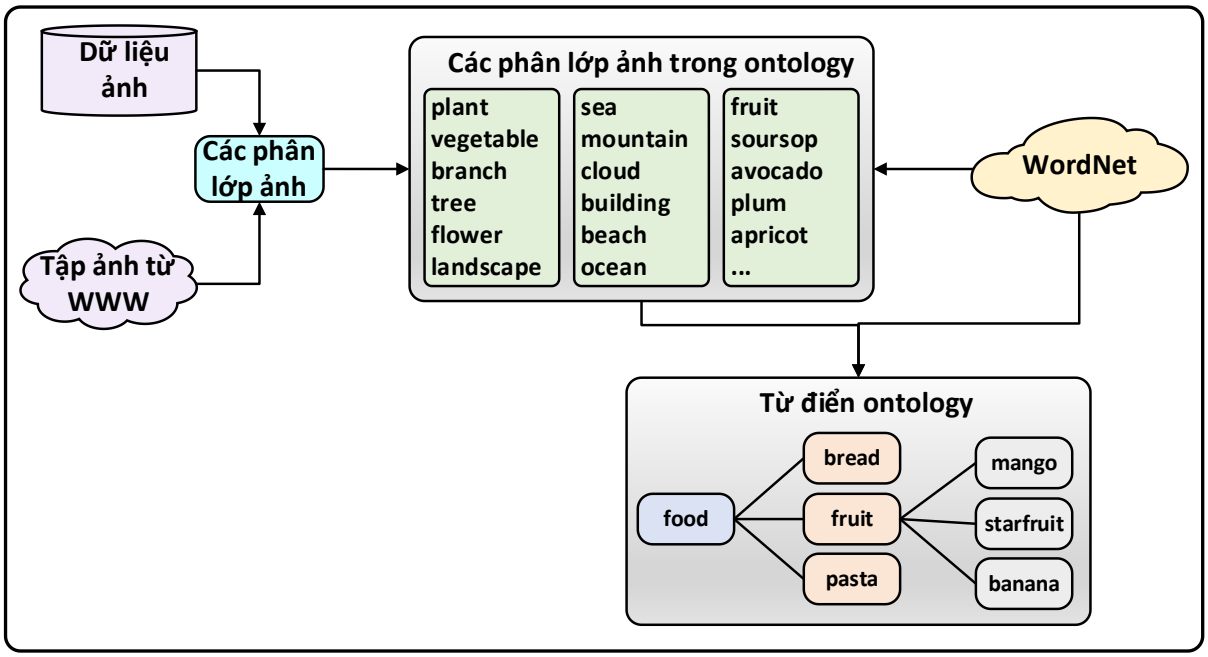
❖ ***Xây dựng khung ontology bán tự động cho tập dữ liệu ảnh***

Dựa trên mô hình khung ontology trong **Hình 2.9**, một khung ontology bán tự động được xây dựng như mô tả trong **Hình 2.11**, bao gồm các bước sau: sử dụng tập dữ liệu ảnh ban đầu và ảnh từ WWW làm dữ liệu đầu vào cho quá trình học ontology; phân lớp tự động các thể từ bộ dữ liệu bằng mô hình học máy như GP-Tree/Graph-GPTree/SgGP-Tree; tạo định nghĩa lớp thủ công hoặc tự động từ WordNet; ảnh từ WWW được gán định danh và URL, bổ sung vào dữ liệu đầu vào; dữ liệu được chỉnh sửa và xác thực bởi các chuyên gia để đảm bảo độ chính xác; khung ontology được tạo bán tự động từ dữ liệu chuẩn hóa.

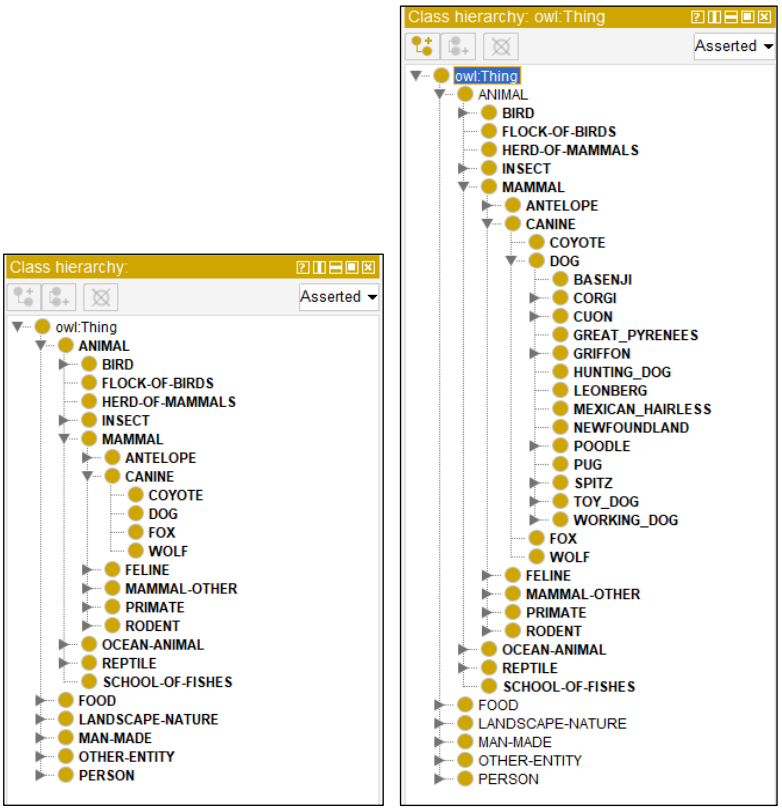


Hình 2.11. Mô hình xây dựng khung ontology bán tự động

Tất cả dữ liệu đều tuân theo cấu trúc cơ bản của một khung ontology chuẩn và đồng thời làm giàu ngữ nghĩa cho ontology. Vì tập dữ liệu phát triển liên tục, khung ontology có thể mở rộng để bổ sung các cá thể hình ảnh, thuộc tính và phân lớp mới. Định nghĩa được lấy từ từ điển ngữ nghĩa WordNet. **Hình 2.12** minh họa việc bổ sung khái niệm mới vào từ điển ontology. Các ảnh từ WWW hoặc dữ liệu ảnh được phân lớp và đối chiếu với các lớp trong ontology; nếu lớp đã có, không bổ sung thêm khái niệm; nếu lớp mới được xác định, định nghĩa được lấy từ WordNet để làm phong phú thêm ngữ nghĩa của ontology. **Hình 2.13** là ví dụ về phân cấp lớp ảnh trước và sau khi làm giàu ontology.



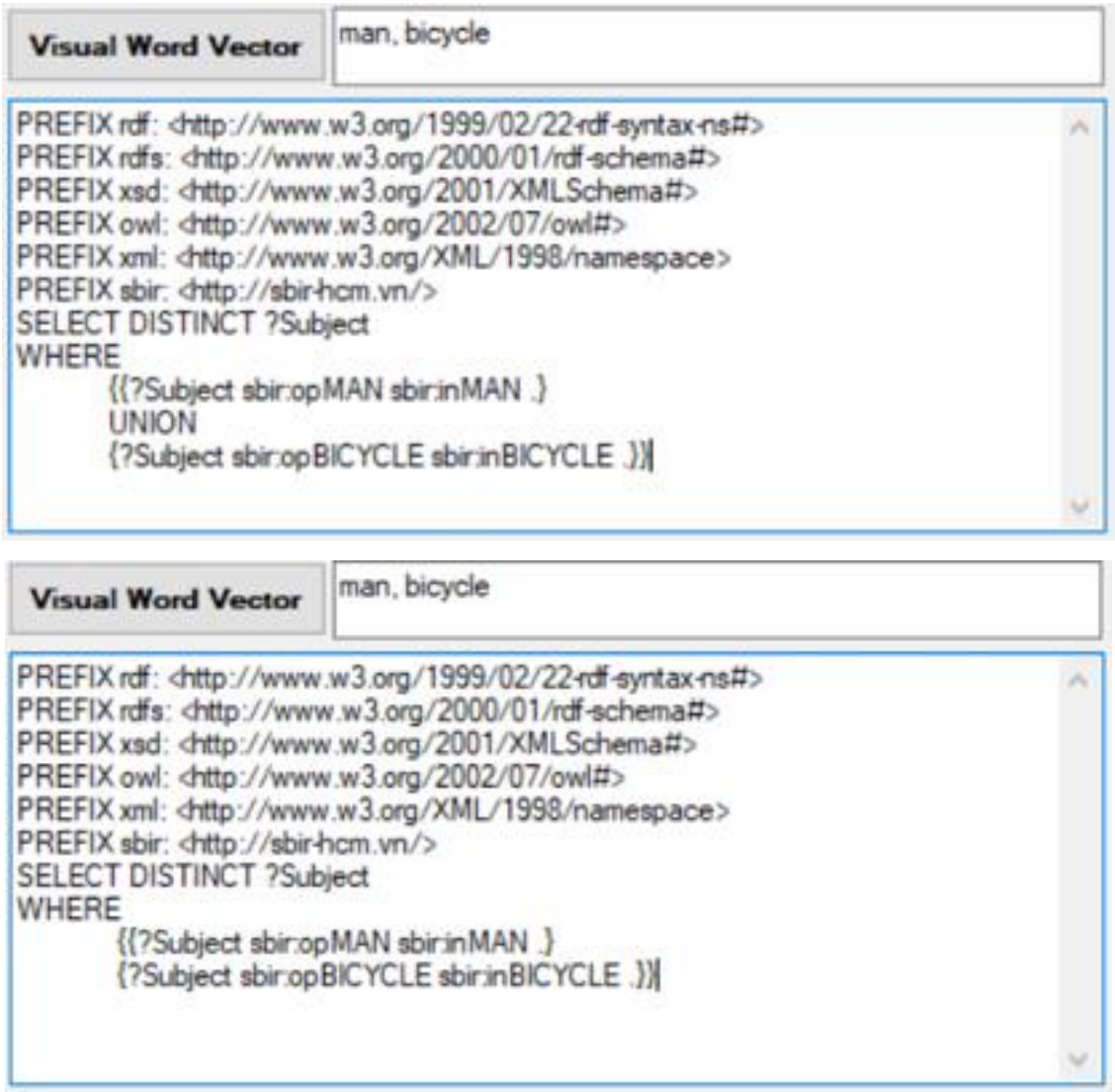
Hình 2.12. Bổ sung khái niệm cho phân lớp mới vào từ điển ontology



Hình 2.13. Ví dụ về ontology trước và sau khi làm giàu

e. Truy vấn SPARQL

Trong phần này, câu truy vấn SPARQL được xây dựng bởi hai phép toán hội (UNION), phép toán và (AND) dựa trên tên phân lớp của ảnh truy vấn đã phân lớp. **Hình 2.14** minh họa một ví dụ về câu truy vấn SPARQL.



Hình 2.14. Câu lệnh SPARQL được tạo dựa trên tập từ vựng thị giác

Hình 2.15. Tập hình ảnh minh hoạ cho truy vấn SPARQL

Bằng cách đưa ra một hình ảnh làm truy vấn, công cụ này sẽ tìm kiếm với tập các ảnh được gắn nhãn hiện có. Hình ảnh truy vấn phải được đặt trong số các danh mục hiện có. Truy vấn có thể ở dạng hình ảnh, dựa trên văn bản hoặc kết hợp của hai hoặc nhiều danh mục, giai đoạn tạo câu truy vấn sẽ được thực hiện tự động. Cho dữ liệu mẫu được mô tả như **Hình 2.15**, các câu truy vấn SPARQL nhằm tìm kiếm các ảnh được trình bày trong **Bảng 2.2.**

Bảng 2.2. Ví dụ truy vấn SPARQL

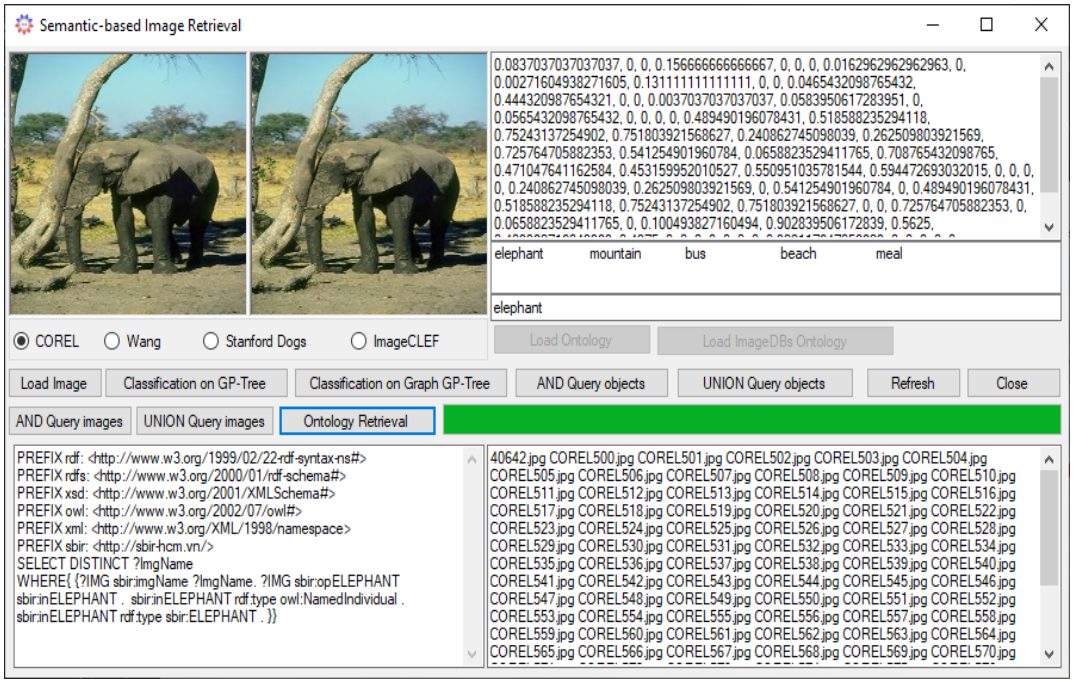
STT	Yêu cầu	Truy vấn	Kết quả
1	Tìm kiếm các phân lớp con của **Bird**	SELECT ?x WHERE { ?x rdfs:subClassOf :Bird }	Duck, Eagle, Peacock, Swan
2	Tìm kiếm các hình ảnh của **Bird**	SELECT ?x ?y WHERE { ?x rdfs:subClassOf :Bird . ?y rdf:type ?x }	
3	Tìm các hình ảnh của phân lớp **Bird** có màu trắng	SELECT ?y ?x WHERE { ?x rdfs:subClassOf :Bird . ?y :Color "White" }	
4	Tìm người tạo các ảnh **Swan**	SELECT DISTINCT ?name WHERE { ?x rdf:type :Swan . ?x :Creator ?name }	Magesh Aswinth
5	Tìm các ảnh về **Swan**	SELECT ?x WHERE { ?x rdf:type :Swan }	
6	Tìm các hình ảnh Car có màu **Red**	SELECT ?x WHERE { ?x rdf:type :Swan }	
7	Tìm các ảnh được tạo bởi **MrAswinth**	SEL ECT ?x ?IName WHERE { ?x :Creator ?IName ; FILTER regex(str(?IName), "Aswinth") }	

2.5.2. Thực nghiệm và đánh giá hệ tìm kiếm GP-SBIR

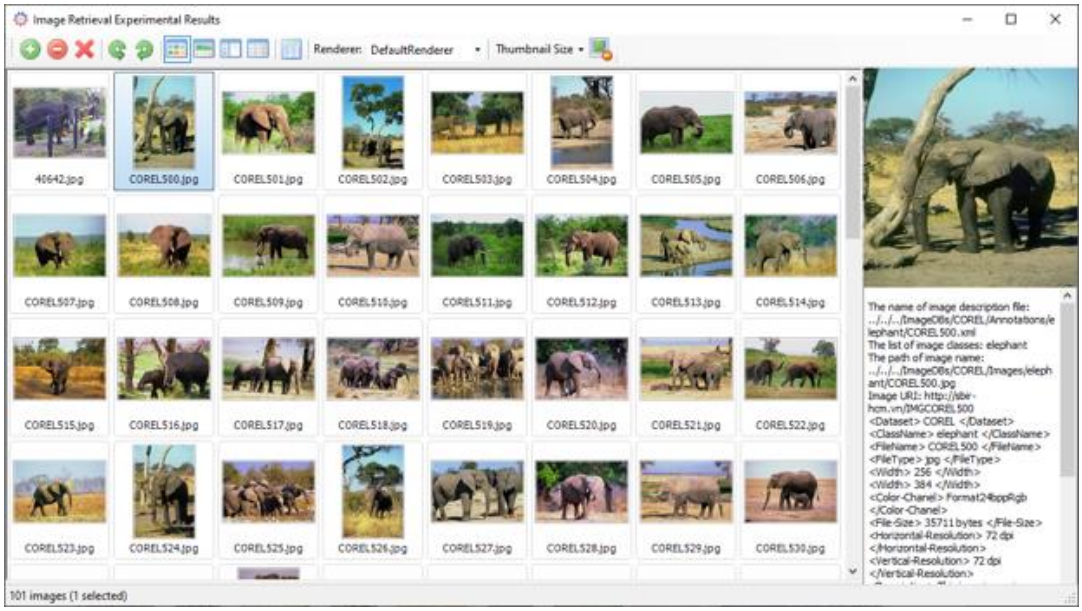
a. Thực nghiệm hệ tìm kiếm ảnh GP-SBIR

Để đánh giá tính chính xác và hiệu quả của lý thuyết đề xuất, thực nghiệm được thực hiện trên ba bộ ảnh đơn đối tượng: WANG (10.800 ảnh), ImageCLEF (20.015 ảnh), và MS-COCO (163.957 ảnh). Hệ thống tìm kiếm ảnh dựa trên GP-Tree (GP-SBIR) được mô tả trong **Hình 2.16**. Đối với mỗi ảnh đầu vào I_q từ các bộ dữ liệu này, véc-tơ đặc trưng được trích xuất và tìm kiếm trên GP-Tree để xác định các ảnh tương tự về nội dung. Kết quả trả về là tập ảnh tương tự SI của ảnh I_q, được xây dựng từ các ảnh cùng phân lớp có tần suất xuất hiện cao nhất. Câu truy vấn SPARQL (Create SPARQL) sau

đó được tạo ra để tìm kiếm các ảnh tương tự theo ngữ nghĩa từ ontology đã được tải (Load Ontology). **Hình 2.17**minh họa kết quả tìm kiếm ảnh tương tự theo ngữ nghĩa.



Hình 2.16. Hệ tìm kiếm ảnh GP-SBIR



Hình 2.17. Kết quả tập ảnh tương tự của ảnh truy vấn trên hệ GP-SBIR

b. Đánh giá thực nghiệm

Thực nghiệm được tiến hành trên ba bộ dữ liệu: WANG, ImageCLEF, và MS-COCO. Vì mỗi bộ dữ liệu có đặc điểm và quy mô khác nhau, cấu trúc cây GP-Tree được tối ưu với các tham số M (số phần tử tối đa ở nút lá) và N (số phần tử tối đa ở nút trong). **Bảng 2.3** trình bày kết quả thực nghiệm với các giá trị M và N đã được điều chỉnh cho từng bộ dữ liệu. Thời gian thực nghiệm ngắn nhất là với bộ ảnh WANG (17.839,47 giây) nhờ số lượng ảnh và cụm ít, trong khi bộ MS-COCO có thời gian thực nghiệm lâu nhất (158.765,84 giây) do quy mô lớn hơn, dẫn đến nhiều lần tách nút và cụm.

Bảng 2.3 cũng thể hiện số lượng ảnh đạt độ chính xác cao nhất (top precision - P@). Các chỉ số đánh giá như Độ chính xác, Độ phủ, Độ dung hòa, và thời gian tìm kiếm (tính bằng milli giây) được sử dụng để đánh giá hiệu quả tìm kiếm. Các chỉ số này, cùng với thời gian tìm kiếm trung bình của từng bộ dữ liệu (WANG, ImageCLEF, MS-COCO), được tổng hợp trong **Bảng 2.4**.

Bảng 2.3. Kết quả thực nghiệm cây GP-Tree

Tên tập ảnh	Số lượng ảnh	Tham số thực nghiệm		Thời gian thực nghiệm (giây)	Số cụm nút lá	Số cụm nút trong	Số mẫu lấy P@	Tỉ lệ P@
		M	N					
WANG	10.800	100	40	17.839,47	218	32	2.240	20%
ImageCLEF	20.000	150	50	32.173,95	432	67	6.000	20%
MS-COCO	163.957	300	70	158.765,84	782	314	44.188	20%

Bảng 2.4. Hiệu suất tìm kiếm ảnh của hệ GP-SBIR trên các tập dữ liệu thử nghiệm

Tập dữ liệu ảnh	Chỉ số đánh giá			
	Độ chính xác	Độ phủ	Độ dung hòa	Thời gian tìm kiếm trung bình (ms)
WANG	0.6780	0.684	0.6810	98.75
ImageCLEF	0.6802	0.775	0.7245	132.09
MS-COCO	0.7170	0.724	0.7205	217.65

Bảng 2.5. So sánh độ chính xác giữa các phương pháp trên bộ dữ liệu WANG

Phương pháp	Độ chính xác trung bình
Bella & Vasuki, 2019 [64]	0.5970
P. Chhabra và cộng sự, 2020 [65]	0.6320
K. Kanwal và cộng sự, 2020 [66]	0.5067
S. Dhingra and P. Bansal, 2021 [67]	0.6000
GP-SBIR	**0.6780**

Bảng 2.6. So sánh độ chính xác giữa các phương pháp trên bộ dữ liệu ImageCLEF

Phương pháp	Độ chính xác trung bình
Y. Qiang và cộng sự, 2020 [68]	0.6670
X. Yue và cộng sự, 2021 [69]	0.67140
N. T. U. Nhi và cộng sự, 2022 [70]	0.6510
X. Wang và cộng sự, 2023 [71]	0.6727
GP-SBIR	**0.6802**

Bảng 2.7. So sánh độ chính xác giữa các phương pháp trên bộ dữ liệu MS-COCO

Phương pháp	Độ chính xác trung bình
Wang, Yang và cộng sự, 2016 [72]	0.6120
Wang, liu và cộng sự, 2019 [73]	0.6890
Wen, S. và cộng sự, 2020 [74]	0.8110
Zhiwei Zhang và cộng sự, 2021 [75]	0.7640
GP-SBIR	**0.7170**

Để đánh giá hiệu quả của hệ thống tìm kiếm ảnh GP-SBIR, kết quả thực nghiệm được so sánh với các phương pháp khác sử dụng cùng bộ dữ liệu. Độ chính xác trung bình của hệ thống được đối chiếu với các phương pháp khác, với kết quả trình bày trong **Bảng 2.5, 2.6** và **2.7**. Các kết quả cho thấy phương pháp GP-SBIR đạt độ chính xác cao khi so với các hệ thống tìm kiếm ảnh ngữ nghĩa khác, và có kết quả tương đương với các phương pháp sử dụng đặc trưng cấp thấp như màu sắc, hình dạng và kết cấu, cùng các thuật toán phân cụm và phân lớp truyền thống.

Tuy nhiên, khi so với các phương pháp học sâu, GP-SBIR chưa đạt được độ chính xác cao. Mặc dù trích xuất đặc trưng thủ công không cho độ chính xác tối ưu, phương pháp này lại đơn giản và nhanh chóng hơn. Điều này chứng tỏ rằng việc kết hợp trích xuất đặc trưng và tổ chức dữ liệu trong cây GP-Tree là hiệu quả đối với các bộ dữ liệu đơn đối tượng. Tuy nhiên, để cải thiện độ chính xác, đặc biệt là đối với các bộ dữ liệu đa đối tượng như ImageCLEF và MS-COCO, cần có sự cải tiến trong cấu trúc cây GP-Tree.

2.6. Tiểu kết chương

Chương này đã giới thiệu cấu trúc cây phân cụm GP-Tree, một giải pháp hiệu quả cho việc lưu trữ và truy xuất dữ liệu lớn, đặc biệt trong tìm kiếm ảnh. GP-Tree áp dụng phương pháp phân cụm phân cấp, giúp tăng tốc tìm kiếm bằng cách duyệt qua các nhánh tương tự nhất. Tại các nút lá, hệ thống xác định những phần tử tương tự, tối ưu thời gian tìm kiếm và đạt độ chính xác khá tốt.

Tuy nhiên, GP-Tree cũng có một số hạn chế. Một trong những nhược điểm chính là trong quá trình phân tách các nút, các phần tử tương tự có thể bị phân ra các nhánh khác nhau, thậm chí có thể nằm ở các nhánh riêng biệt, dẫn đến việc bỏ sót các phần tử tương tự trong quá trình tìm kiếm, làm giảm hiệu suất, đặc biệt khi các đối tượng tương tự không còn nằm gần nhau.

Chương tiếp theo sẽ giới thiệu các cải tiến cho GP-Tree nhằm nâng cao độ chính xác trong tìm kiếm ảnh, tập trung vào việc tối ưu cấu trúc cây để giảm thiểu tình trạng bỏ sót phần tử tương tự. Ngoài ra, một mô hình tìm kiếm ảnh ngữ nghĩa dựa trên ontology cũng sẽ được đề xuất để cải thiện hiệu suất tìm kiếm.

CHƯƠNG 3. CẤU TRÚC SGGP-TREE ĐỂ TÌM KIẾM ẢNH THEO NGỮ NGHĨA

Chương này trình bày các cải tiến đối với cấu trúc cây phân cụm GP-Tree nhằm tối ưu hóa hiệu quả tìm kiếm ảnh. Các phương pháp như Graph-GPTree và mạng kết hợp SgGP-Tree được đề xuất để cải thiện khả năng lưu trữ và truy xuất phần tử tương tự. Đồng thời, phương pháp tìm kiếm ảnh ngữ nghĩa dựa trên ontology được thảo luận, với SgGP-Tree giúp phân lớp đối tượng chính xác hơn. Một mô hình tìm kiếm ảnh ngữ nghĩa kết hợp giữa ontology và SgGP-Tree cũng được thử nghiệm trên các bộ dữ liệu như WANG, MS-COCO và ImageCLEF để đánh giá hiệu quả. Các kết quả này đã được công bố trong các công trình [CT1], [CT2], [CT3], [CT6].

3.1. Giới thiệu

Cây GP-Tree được trình bày ở **Chương 2** là một cấu trúc dữ liệu được xây dựng để tổ chức và quản lý các tập dữ liệu lớn, đặc biệt là dữ liệu hình ảnh, nhằm tối ưu hóa hiệu suất tìm kiếm và truy xuất thông tin. Cây GP-Tree dựa trên nguyên tắc phân cụm và phân cấp, cho phép chia nhỏ các tập dữ liệu thành các nhánh cây, mỗi nhánh đại diện cho một cụm dữ liệu. Nhờ đó, việc tìm kiếm các phần tử trong tập dữ liệu trở nên nhanh chóng và hiệu quả hơn so với các cấu trúc dữ liệu truyền thống như cây nhị phân hay danh sách liên kết.

Ưu điểm của cây GP-Tree là (1) tối ưu hóa không gian: giúp giảm thiểu dung lượng bộ nhớ cần thiết để lưu trữ dữ liệu, nhờ vào việc tổ chức dữ liệu theo cách phân cụm và phân cấp; (2) tăng tốc độ truy vấn: việc tổ chức dữ liệu theo cây giúp giảm số lượng so sánh cần thiết trong quá trình tìm kiếm, từ đó rút ngắn thời gian truy vấn; (3) quản lý dữ liệu: GP-Tree có thể dễ dàng mở rộng để xử lý các tập dữ liệu đa dạng và phức tạp.

Tuy nhiên, GP-Tree cũng gặp phải một số hạn chế, đặc biệt trong quá trình xây dựng cây. Khi một nút lá chứa số lượng phần tử vượt quá giới hạn cho phép (M), thuật toán sẽ tách nút để đảm bảo mỗi nút không vượt quá M phần tử. Tuy nhiên, quá trình này có thể gây ra các vấn đề như phân tán dữ liệu, tăng chi phí tính toán và làm phức tạp hóa việc tìm kiếm.

Khi một nút lá chứa quá nhiều phần tử, cây sẽ tách nút này thành hai hoặc nhiều nút mới. Tuy nhiên, trong quá trình tách này, các phần tử tương tự nhau có thể bị phân tách vào các nhánh khác nhau. Điều này làm cho việc tìm kiếm các phần tử tương tự trở nên khó khăn hơn, do hệ thống không thể liên kết các phần tử đó với nhau. Ngoài ra, Các phương pháp tìm kiếm ảnh truyền thống thường dựa vào việc so sánh từng phần tử trong tập dữ liệu với nhau, dẫn đến chi phí tính toán cao và thời gian tìm kiếm dài. Điều này sẽ thể hiện rõ ràng trong các tập dữ liệu lớn.

Để cải thiện hiệu suất truy vấn và khắc phục các vấn đề nêu trên, việc phát triển các phương pháp mới dựa trên cây GP-Tree là rất cần thiết. Nhiều nghiên cứu đã chỉ ra rằng các phương pháp tìm kiếm ảnh dựa trên đồ thị có thể cung cấp giải pháp tối ưu cho các vấn đề này như tìm kiếm ảnh sử dụng k-NN, xếp hạng dựa trên đồ thị, cấu trúc đồ thị cụm chữ ký nhị phân. Mặc dù các phương pháp này đã chứng minh hiệu quả trong nhiều ứng dụng, nhưng chúng thường đòi hỏi thời gian tìm kiếm và độ phức tạp cao do phải quản lý một lượng lớn cụm. Điều này có thể dẫn đến chi phí tính toán cao, làm giảm hiệu suất tổng thể của hệ thống.

Nhằm khắc phục những nhược điểm trong cấu trúc GP-Tree và cải thiện hiệu suất tìm kiếm, một cấu trúc mới được đề xuất kết hợp giữa cây GP-Tree và đồ thị cụm lân cận, gọi là Graph-GPTree. Cấu trúc này có những điểm nổi bật như:

- Kết nối các phần tử tương tự: Graph-GPTree tạo ra một cấu trúc đồ thị từ các nút lá, giúp liên kết các phần tử tương tự với nhau, bất kể chúng có thể nằm ở các nhánh khác nhau trong cây. Điều này cho phép truy vấn hình ảnh tìm kiếm được các đối tượng tương tự mà không bỏ sót.
- Tránh bỏ sót thông tin: đồ thị cụm lân cận giúp tăng cường khả năng truy xuất các phần tử tương tự, từ đó cải thiện độ chính xác trong truy vấn. Nhờ đó, người dùng có thể tìm thấy các phần tử liên quan mà không gặp phải các vấn đề phân tán do quá trình tách nút.
- Giảm số lượng so sánh: Bằng cách tổ chức các phần tử tương tự thành các nút lá liên kết trong một đồ thị, Graph-GPTree giúp giảm số lượng so sánh cần thiết

trong quá trình tìm kiếm. Người dùng chỉ cần tìm kiếm trong đồ thị để xác định các phần tử tương tự thay vì phải quét qua toàn bộ cây.

- Tăng tốc độ truy vấn: Đồ thị cụm lân cận cho phép thực hiện các phép toán tìm kiếm nhanh hơn thông qua việc sử dụng các chỉ số gần nhất (k-NN), giúp cải thiện thời gian phản hồi cho người dùng.

Mặc dù đồ thị cụm lân cận giúp giảm thiểu tình trạng bỏ sót thông tin, việc chọn cụm dựa trên độ đo vẫn có thể dẫn đến sai sót. Nếu cây GP-Tree tách nút quá nhiều lần mà không có chiến lược hợp lý, việc phân tán dữ liệu có thể làm giảm độ chính xác của tìm kiếm. Mạng tự tổ chức (SOM) là một phương pháp học không giám sát có khả năng tổ chức và phân nhóm dữ liệu một cách tự động. Việc áp dụng SOM vào cây GP-Tree giúp giải quyết một số vấn đề còn tồn tại trong cấu trúc cây như:

- Tự động tổ chức dữ liệu: SOM có khả năng tự tổ chức các phần tử hình ảnh dựa trên các đặc điểm tương đồng. Điều này giúp giảm thiểu tình trạng phân tán và tổ chức lại các phần tử vào các cụm hợp lý.
- Giảm thiểu sai sót khi chọn cụm: Mạng SOM giúp chọn cụm chiến thắng một cách hiệu quả hơn dựa trên các tiêu chí đã định, từ đó cải thiện độ chính xác trong việc xác định các phần tử tương tự. Điều này đặc biệt hữu ích khi có nhiều cụm và cần thiết phải phân loại chính xác các phần tử.

Do đó, khi kết hợp Graph-GPTree và mạng SOM, các vấn đề tồn tại trong cây GP-Tree được giải quyết một cách hiệu quả hơn. Dựa trên những phân tích và giải pháp nêu trên, luận án đề xuất cải tiến cây GP-Tree theo các hướng chính như sau:

1. Tạo cấu trúc Graph-GPTree: Đồ thị này được hình thành từ các lân cận của nút lá trong cây GP-Tree. Trong mỗi lần tách nút, cây GP-Tree sẽ thực hiện đánh dấu lân cận cho các nút lá mới tách ra theo tiêu chí xác định. Điều này giúp đảm bảo rằng các phần tử tương tự không bị bỏ sót trong quá trình truy vấn, nâng cao độ chính xác của tìm kiếm.
2. Tạo cấu trúc SgGP-Tree: Đây là một mạng SOM được lắp ghép từ đồ thị cụm lân cận Graph-GPTree. Cấu trúc này được xây dựng dựa trên bộ véc-tơ trọng số được

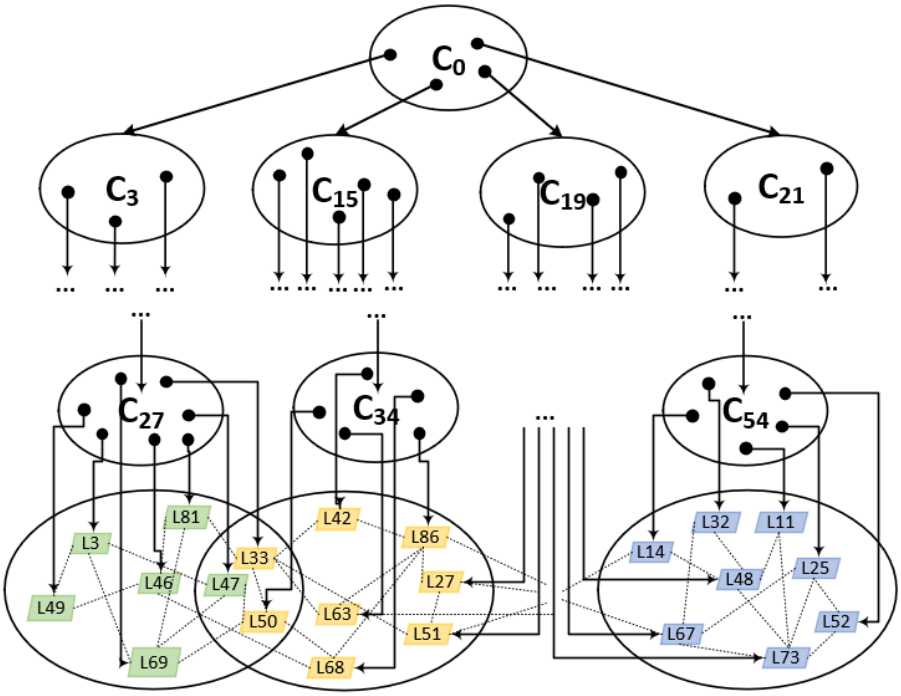
huấn luyện trên cây GP-Tree, nhằm tìm cụm chiến thắng theo phân lớp đại diện đã đề xuất. Việc kết hợp này không chỉ giúp nâng cao hiệu quả tìm kiếm mà còn đảm bảo rằng các phần tử tương tự được tổ chức và quản lý hiệu quả hơn.

Chương này bao gồm các nội dung chính như sau: Mục 3.2 giới thiệu về cấu trúc đồ thị cụm Graph-GPTree; cấu trúc mạng kết hợp SgGP-Tree được trình bày trong Mục 3.3; Hệ tìm kiếm ảnh theo ngữ nghĩa dựa trên SgGP-Tree và đánh giá kết quả thực nghiệm được mô tả chi tiết ở Mục 3.4. Cuối chương, phần tiểu kết được nêu trong Mục 3.5.

3.2. Đồ thị cụm Graph-GPTree

3.2.1. Cấu trúc Graph-GPTree

Graph-GPTree được tạo dựa trên các thao trên đồ thị thưa tập các *nút lá* thu được của GP-Tree. Trong đó, các đỉnh biểu thị các *nút lá* và các cạnh có trọng số biểu thị mức độ tương đồng giữa chúng. Đồ thị thưa được tạo trong quá trình tạo cây GP-Tree khi mỗi lần tách nút lá hệ thống tiến hành đánh dấu các mức lân cận của các *nút lá* vừa mới tách. **Hình 3.1**Error! Reference source not found. mô tả tổng quan việc tạo đồ thị thưa dựa trên tập nút lá cây GP-Tree.

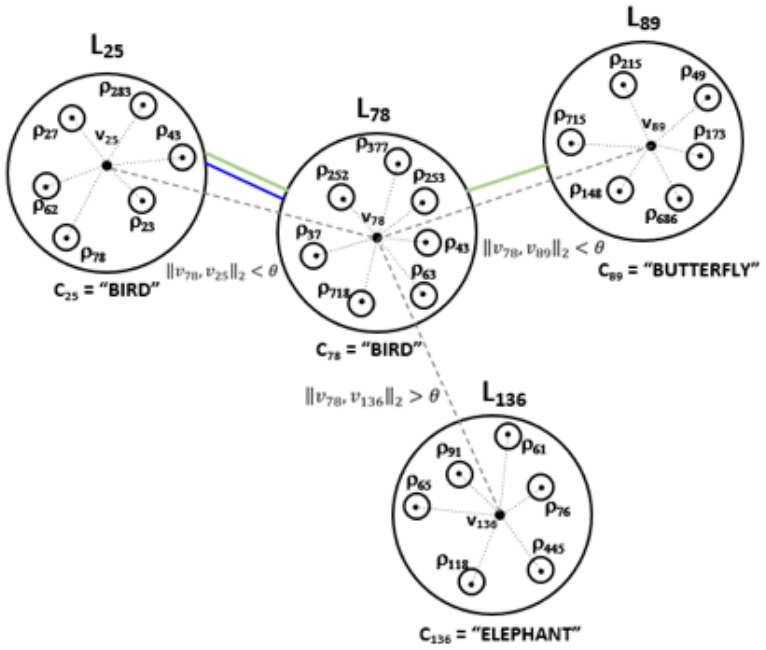


Hình 3.1. Đồ thị thưa được tạo phải tập nút lá cây GP-Tree

Các cấp lân cận giữa hai *nút lá* bất kỳ L_i và L_j được xác định như sau:

Định nghĩa 3.1: *Đồ thị cụm lân cận*

- *Lân cận cấp 1: Gọi* $v_p = (v_1^p, v_2^p, \dots, v_n^p)$, $v_q = (v_1^q, v_2^q, \dots, v_n^q)$ *là các véc-tơ tâm của hai nút lá* L_p *và* L_q, *với* $v_j^p = \sum_{i=1}^{m_p} f_{ij}^p, \forall j = \overline{1,n}$; $v_j^q = \sum_{i=1}^{m_q} f_{ij}^q, \forall j = \overline{1,n}$. *Nếu* $\|v_p, v_q\|_2 < \theta$, *với* θ *là một giá trị ngưỡng cho trước, thì* L_p, L_q *được đánh dấu là lân cận cấp 1 với nhau.*
- *Lân cận cấp 2: Gọi* r, s *lần lượt là số nhãn phân lớp của ảnh xuất hiện trong hai nút lá* L_t *và* L_k; c_t, c_k *là nhãn phân lớp xuất hiện nhiều nhất trong hai nút lá đó, trong đó:* $c_t = argmax\{count(\eta_i.c_j) | \eta_i \in L_t, i = 1..|L_t|, j = 1..r\}$, $c_k = argmax\{count(\eta_i.c_j) | \eta_i \in L_k, i = 1..|L_k|, j = 1..s\}$. *Nếu* $c_t \equiv c_k$, *thì* L_t *và* L_k *được đánh dấu là lân cận cấp 2 với nhau.*



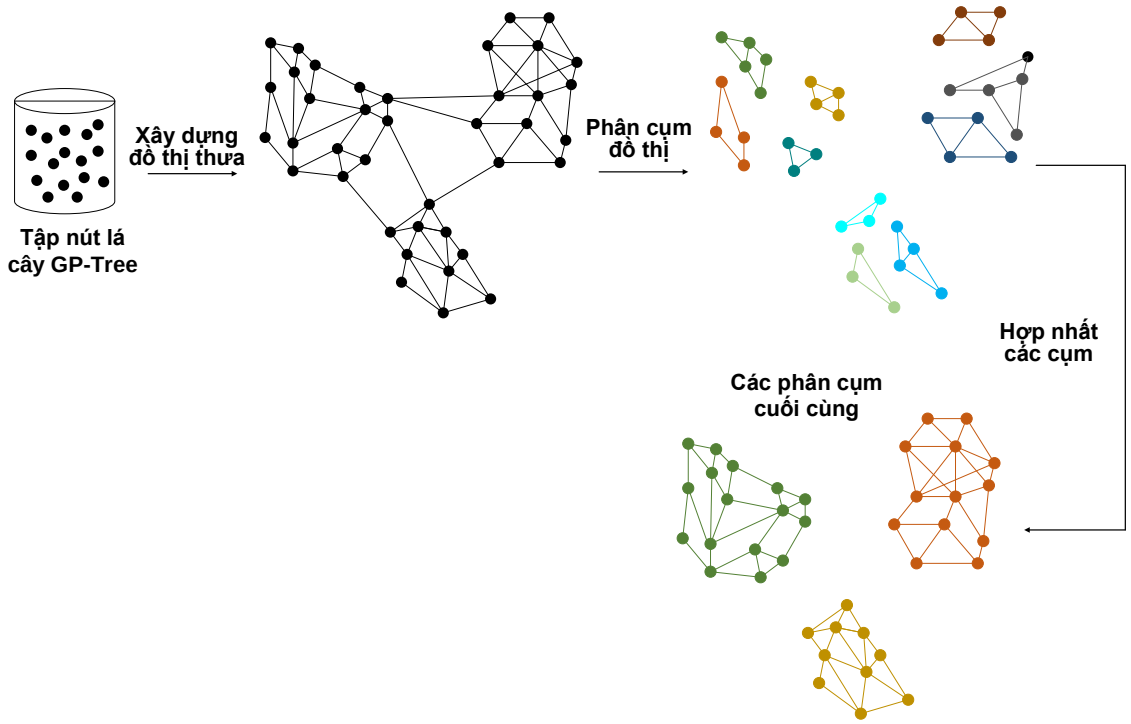
Hình 3.2. Ví dụ về đồ thị cụm lân cận của nút lá $\boldsymbol{L_{78}}$

Hình 3.2 là đồ thị cụm lân cận như **Định nghĩa 3.1** từ nút L_{78}. Lớp đại diện của lá này là $\mu_{78} = "BIRD"$. Các cấp lân cận được thể hiện bằng các đường liên kết, với cấp 1 là đường kẻ liền màu xanh lá cây đậm và cấp 2 là đường màu xanh dương. Qua đó, ta có:

- Do $\|v_{78}, v_{25}\|_2 < \theta$ nên L_{78} là lân cận cấp 1 của L_{25}. Đồng thời lớp đại diện của L_{25} là $\mu_{25} =$ "BIRD", do đó $\mu_{25} \equiv C_{78}$, nên L_{25} cũng là lân cận cấp 2 của L_{78};
- Do $\|v_{78}, v_{89}\|_2 < \theta$ nên L_{89} là lân cận mức 1 của L_{78};
- Do $\|v_{78}, v_{136}\|_2 > \theta$ và lớp đại diện của L_{136} là $\mu_{136} =$ "ELEPHANT" nên L_{136} không là lân cận của L_{78} .

3.2.2. Quá trình tạo Graph-GPTree

Thuật toán phân cụm đồ thị GraphGP-Tree để tìm các cụm trong một tập dữ liệu được mô tả tổng quát như trong **Hình 3.3**. Thuật toán thực hiện trên một đồ thị thưa trong đó các nút biểu thị các phần tử dữ liệu và các cạnh có trọng số biểu thị sự tương đồng giữa các phần tử dữ liệu. Việc biểu diễn tập dữ liệu bằng đồ thị thưa này cho phép thuật toán phân cụm mở rộng quy mô thành các tập dữ liệu lớn. Thuật toán tìm các cụm trong tập dữ liệu có hai giai đoạn: (1) Trong giai đoạn thứ nhất, thuật toán sử dụng giải thuật phân vùng đồ thị dựa trên phương pháp cắt đồ thị để phân cụm các phần tử dữ liệu thành một số lượng lớn các cụm con tương đối nhỏ. (2) Trong giai đoạn thứ hai, sử dụng thuật toán phân cụm theo phân cấp hợp nhất để tìm các cụm chính bằng cách liên tục kết hợp các cụm con này lại với nhau.



Hình 3.3. Tạo đồ thị phân cụm dựa trên tập nút lá của GP-Tree

❖ **Giai đoạn thứ nhất:**

Giai đoạn này tìm các cụm con ban đầu bằng thuật toán phân vùng đồ thị để phân vùng đồ thị k lân cận gần nhất của tập dữ liệu thành một số lượng lớn các phân vùng sao cho đường cắt cạnh nhỏ hơn ngưỡng θ cho trước. Vì mỗi cạnh trong biểu đồ lân cận k gần nhất thể hiện sự giống nhau giữa các điểm dữ liệu, nên dữ liệu trong mỗi phân vùng có mối liên quan chặt chẽ với các mục dữ liệu khác trong cùng phân vùng. Các bước thực hiện việc xác định các cụm con như sau: (1) Sắp xếp các cạnh theo trọng số tăng dần; (2) Sắp xếp các đỉnh (dòng và cột) theo trọng số tăng dần; (3) Sử dụng ngưỡng θ để tạo ra các phân cụm dựa trên kỹ thuật cắt đồ thị.

❖ **Giai đoạn thứ hai:**

Hợp nhất các cụm con bằng cách sử dụng phân cụm phân cấp hợp nhất để kết hợp các cụm con nhỏ được xác định trong tiến trình thứ nhất lại với nhau. Bước quan trọng của thuật toán phân cụm hợp nhất là tìm ra cặp cụm con giống nhau nhất dựa trên ngưỡng θ. Ngưỡng này dùng để kiểm soát tính đồng nhất về độ giống nhau giữa các mục dữ liệu thuộc một cụm cụ thể.

Lặp lại hai tiến trình trên cho đến khi chỉ còn một phân cụm duy nhất hoặc không mới được tạo ra khi tiến hành hợp nhất các cụm con. Tại thời điểm này, thuật toán tạo Graph-GPTree sẽ kết thúc và đưa ra các phân cụm hiện tại làm kết quả.

Việc biểu diễn các mục dữ liệu bằng biểu đồ thưa của GraphGP-Tree dựa trên cách tiếp cận biểu đồ k-NN thường được sử dụng. Mỗi đỉnh của biểu đồ lân cận gần nhất k đại diện cho một mục dữ liệu và tồn tại một cạnh giữa hai đỉnh, nếu các mục dữ liệu tương ứng với một trong các nút nằm trong số k điểm dữ liệu tương tự nhất của điểm dữ liệu tương ứng với nút kia theo một ngưỡng θ cho trước.

Đồ thị Graph-GPTree $G = (V, E)$ được phân cụm theo các bước như sau:

(1) Tìm $e_{ij} = \min(E)$, lúc đó 2 đỉnh v_i, v_j được gom thành một cụm

(2) Gọi v_k là đỉnh đại diện cho cụm 2 đỉnh v_i, v_j, khi đó cạnh giữa đỉnh v_k và các đỉnh còn lại được xác định như sau:

$$e_{kt} = max\big(d_E(v_t, v_i), d_E(v_t, v_j)\big), \forall v_t \in V \backslash \{v_i, v_j\}$$

(3) Lập lại quá trình (1),(2) cho đến khi không có cụm mới được tạo ra hoặc chỉ còn một cụm duy nhất thì dừng

Gọi $(m_{ij})_{N_L \times N_L}$ là ma trận biểu diễn đồ thị G, trong đó $m_{ij} = \langle e_{ij}, \tau \rangle$, với $\forall e_{ij} \in E$, τ là chỉ mục của phần tử m_{ij} trong ma trận $(m_{ij})_{N_L \times N_L}$, N_L là tập các nút lá tiến hành phân cụm. **Thuật toán 3.1** phân cụm đồ thị được mô tả như sau:

Thuật toán 3.1: Phân cụm đồ thị Graph-GPTree

Input: $(v_1, \dots, v_{N_L})$
Output: danh sách các cụm con δ
function hc(G, α)
Begin
    For $i \leftarrow 1$ **to** N_L **do**
        For $j \leftarrow 1$ **to** N_L **do**
            $m_{ij}.\omega \leftarrow e_{ij}$
            $m_{ij}.\tau \leftarrow j$
        $I[i] \leftarrow i$
        $\wp[i] \leftarrow$ argmin$\{m_{ik}.\omega \mid i \neq j\}, \forall k = 1..N_L$
    $\delta \leftarrow [\,]$
    For $k \leftarrow 1$ **to** $N_L - 1$ **do**
        $i_1 \leftarrow$ argmin$\{\wp[i].\omega \mid I[i] = i\}, \forall i = 1..N_L$
        $i_2 \leftarrow I[\wp[i_1].\tau]$
        δ.APPEND$(\langle i_1, i_2 \rangle)$
        for $i \leftarrow 1$ to N_L do
            if $I[i] = i \wedge i \neq i_1 \wedge i \neq i_2$ then
                $m_{i_1 i}.\omega \leftarrow m_{i i_1}.w \leftarrow \max(m_{i_1 i}.\omega, m_{i_2 i}.\omega)$
            if $I[i] = i_2$then
                $I[i] \leftarrow i_1$
            $\wp[i_1] \leftarrow$ argmin$\{m_{i_1 i}.\omega \mid I[i] = i \wedge i \neq i_1\}, \forall i = 1..N_L$
    Return δ
End

Tính chất 3.1: Độ phức tạp của **Thuật toán 3.1** là $O({N_L}^2)$.

C h ứ n g m i n h:

Quá trình phân cụm trong **Thuật toán 3.1**, mảng $\wp$ được sử dụng ghi lại cụm hợp nhất tốt nhất cho mỗi cụm. Sau khi hợp nhất hai cụm i_1 và i_2, cụm đầu tiên (i_1) đại diện cho

cụm đã hợp nhất. Nếu $I[i] = i$, thì i là đại diện của cụm hiện tại của nó. Nếu $I[i] \neq i$, thì i đã được hợp nhất vào cụm được đại diện bởi $I[i]$ và do đó sẽ bị bỏ qua khi cập nhật $\wp[i_1]$. Trong thuật toán phân cụm đồ thị, hai vòng lặp *for* cấp cao nhất là $O\left({N_L}^2\right)$, do đó độ phức tạp tổng thể của phân cụm liên kết tối đa là $O\left({N_L}^2\right)$■.

Thuật toán tách nút lá trên cây GP-Tree và tạo đồ thị cụm lân cận Graph-GPTree được trình bày trong **Thuật toán 3.2**. Trong đó, L_s là nút lá cần tách và θ là ngưỡng khoảng cách xác định lân cận cấp một theo **Định nghĩa 3.1** và M là số phần tử tối đa trong một nút lá.

Thuật toán 3.2: Tách nút lá trên GP-Tree, tạo đồ thị cụm lân cận Graph-GPTree

Input: Ngưỡng θ, nút lá L_s;
Output: Đồ thị Graph-GPTree sau khi tách nút lá trên cây GP-Tree
Function: createGraph**(**θ, L_s**)**
Begin
    # Tìm hai phần tử xa nhau nhất của nút lá.
    $c_s = \frac{1}{m_s}\sum_{i=1}^{m_s} \rho_i^s . f_i^s$
    ρ_i^l = argmax$\{\|c_s, \rho_i^s . f\|_2, i = 1..m_k\}$
ρ_j^r=argmax$\left\{\left\|\rho_i^l . f, \rho_i^s . f\right\|_2, , i = 1..m_k\right\}$
    # Tạo hai nút lá mới
    $L_l, L_r \leftarrow$ Khởi tạo các nút lá mới
    $L_l = \{L_l\} \cup \rho_i^l; \quad L_r = \{L_r\} \cup \rho_j^r$
    # Phân bổ phần tử cho 2 nút lá.
    Foreach $\rho_i^s \in L_s$ **do**
        If $\left\|\rho_i^s, \rho_i^l\right\|_2 < \left\|\rho_i^s, \rho_j^r\right\|_2$ **Then**
            $L_l = \{L_l\} \cup \rho_i^s$
        Else
            $L_r = \{L_r\} \cup \rho_i^s$
        EndIf
    EndForeach
    # Tạo phần tử tâm của 2 nút lá: L_l & L_r
    $\sigma_l^h . c^h = \frac{1}{m_l}\sum_{i=1}^{m_l} \rho_i^l . f_i^l ; \sigma_r^h . c^h = \frac{1}{m_r}\sum_{i=1}^{m_r} \rho_i^r . f_i^r$
    #Cập nhật phần tử đại diện cho nút cha
    $\sigma_h^k = \sigma_h^k \cup \left\{\sigma_l^h, \sigma_r^h\right\}$
    # Xác định các lân cận cấp một của nút lá mới
    If $\left\|\sigma_l^h . c^h, \sigma_r^h . c^h\right\|_2 < \theta$ **Then**

26	$\Psi_1.L_l = \Psi_1.L_l \cup \{L_r\}$; $\Psi_1.L_r = \Psi_1.L_r \cup \{L_l\}$
27	**EndIf**
28	# Xác định các lân cận cấp hai của nút lá mới
29	$\gamma_l = \text{argmax}\{\text{count}(\rho_i^l.\mu_i^l), i = 1..\lvert m_l\rvert\}$
30	$\gamma_r = \text{argmax}\{\text{count}(\rho_j^r.\mu_j^r), j = 1..\lvert m_r\rvert\}$
31	**If** $\gamma_l = \gamma_r$ **Then**
32	$\Psi_2.L_l = \Psi_2.L_l \cup \{L_r\}$; $\Psi_2.L_r = \Psi_2.L_r \cup \{L_l\}$;
33	**EndIf**
34	$\text{Graph} - \text{GPTree} = \text{Graph} - \text{GPTree} \cup \{\Psi_1, \Psi_2\}$
35	**Return** Graph-GPTree
36	**End**

Tính chất 3.2: Độ phức tạp của **Thuật toán 3.2** là $O(M)$.

C h ứ n g m i n h:

Độ phức tạp thời gian chính của thuật toán nằm ở vòng lặp "foreach", khi phân bổ các phần tử từ nút lá L_s vào hai nút lá mới. Vì số phần tử tối đa tại một nút lá là M, nên vòng lặp sẽ thực thi M lần để thêm phần tử vào nút mới, dẫn đến độ phức tạp thời gian của thuật toán là $O(M)$ ■.

Quá trình tìm kiếm ảnh trên đồ thị kết hợp hai giai đoạn: tìm kiếm trên cây và tìm kiếm trên đồ thị. Đầu tiên, đặc trưng ảnh truy vấn được trích xuất thành phần tử dữ liệu ρ. Sau đó, ρ được so sánh với các phần tử đại diện σ tại các nút trong của cây, di chuyển từ gốc đến lá theo độ đo Euclid. Khi đạt đến nút lá, ảnh được truy vấn trên đồ thị để tìm các nút lá láng giềng Ω, từ đó thu thập các phần tử Ψ tương tự ρ. Quy trình tìm kiếm này được mô tả trong **Thuật toán 3.3**.

	Thuật toán 3.3: Tìm kiếm ảnh trên đồ thị Graph-GPTree
1	**Input**: r, ρ , G
2	**Output**: Tập các ảnh tương tự Ψ với ảnh truy vấn
3	**Function**: retrieveGraph(r, ρ, G)
4	**Begin**

5 If (r là nút lá) then
6 $\Psi \leftarrow \{\rho_i^r | i = 1..m_r\}$ **d**
7 **Else**
8 $\varphi \leftarrow \text{argmin}\{\|\rho, \sigma_i^r\|_2 | i = 1..n_r\}$
9 **If** φ là nút lá **Then**
10 $\Omega \leftarrow$ lấy các nút lá lân cận trong cụm chứa φ
11 $\Psi \leftarrow \rho_i^{\varphi}$
12 **For** $L_k \in \Omega$ d**o**
13 $\Psi \leftarrow \Psi \cup \{\rho_i^k | i = 1..m_k\}$
14 **EndFor**
15 **Else**
16 retrieveGraph(φ, ρ, G) // Đệ quy
17 **EndIf**
18 **EndIf**
19 **Return** Ψ
20 **End**

Tính chất 3.3: Độ phức tạp của **Thuật toán 3.3** là $O(h \times log(h) \times k)$.

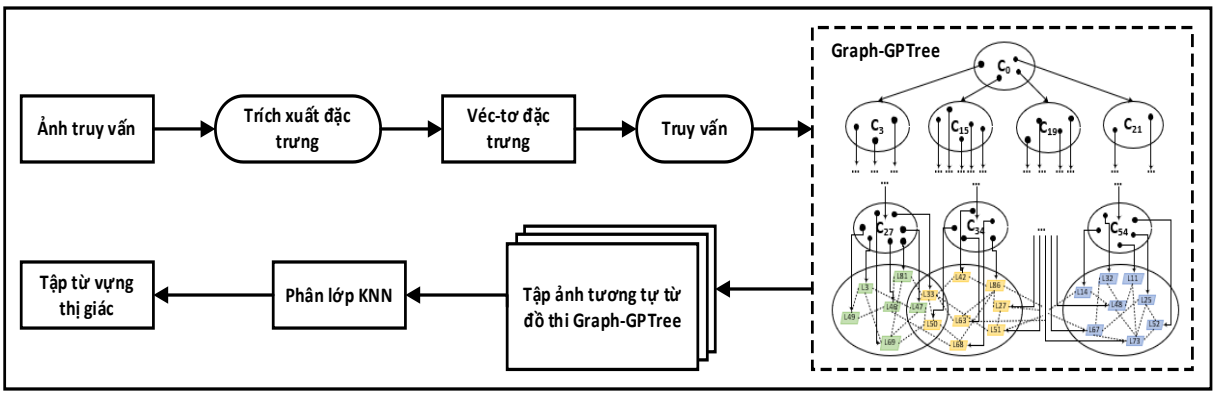
C h ứ n g m i n h:

Độ phức tạp thời gian chính của **Thuật toán 3.3** ở dòng 9 đến dòng 18, đây là bước trích xuất các phần tử dữ liệu trong tập các nút lá. Thuật toán tìm kiếm ảnh trên đồ thị cụm thực hiện gọi đệ quy theo một nhánh của cây để tìm kiếm nút lá phù hợp và trích xuất các phần tử trong nút lá đó, mỗi lần duyệt một nút sẽ kiểm tra phần tử đại diện của nút đó. Nút lá thu được sẽ tiếp tục truy vấn trên đồ thị với k cụm. Vì vậy, Thuật toán tìm kiếm ảnh có độ phức tạp là $O(h \times log(h) \times k)$, với h và k lần lượt là chiều cao của cây GP-Tree và số cụm của đồ thị. Do đó, thuật toán này là khả thi và hữu hạn bước để thực thi■.

3.2.3. Mô hình tìm kiếm ảnh trên Graph-GPTree

Mô hình tìm kiếm ảnh dựa trên đồ thị cụm lân cận Graph-GPTree, như thể hiện trong **Hình 3.4**, thực hiện các bước sau: Trích xuất đặc trưng ảnh đầu vào, so sánh với cơ sở dữ liệu trên cây GP-Tree để chọn nhánh tương tự nhất và xác định nút lá phù hợp. Sau

đó, sử dụng đồ thị cụm Graph-GPTree để tìm các nút lá lân cận và sắp xếp các ảnh tương tự theo độ đo tăng dần.

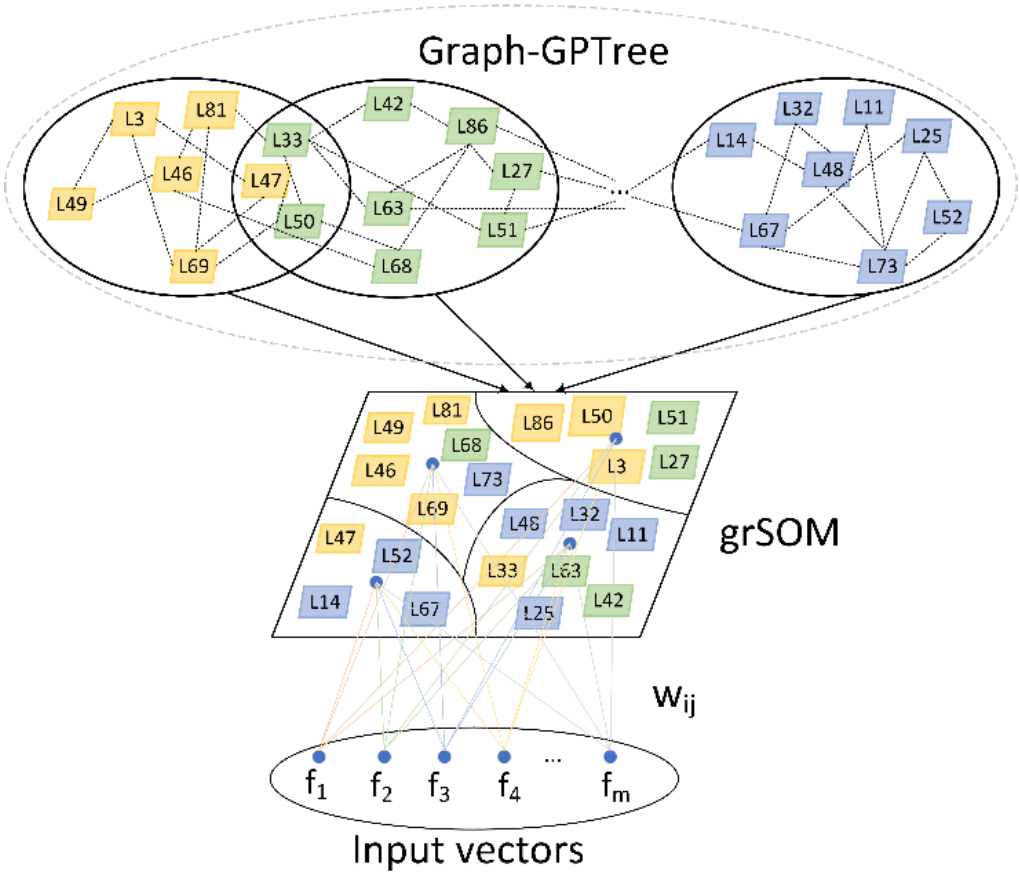


Hình 3.4. Mô hình tìm kiếm ảnh trên đồ thị cụm lân cận Graph-GPTree

3.3. Mạng kết hợp SgGP-Tree

3.3.1. Cấu trúc SgGP-Tree

Mạng kết hợp SgGP-Tree Cấu trúc SgGP-Tree kết hợp cây GP-Tree, Graph-GPTree và mạng SOM [76]. Mạng SOM điều chỉnh trọng số để gom cụm tốt nhất, nhưng chi phí huấn luyện cao và có thể gặp vấn đề khi dữ liệu mới được thêm vào. Để khắc phục, một mạng SOM trên đồ thị Graph-GPTree, gọi là GrSOM, đã được đề xuất. GrSOM sử dụng các nút lá từ đồ thị Graph-GPTree, với các véc-tơ trọng số lấy từ quá trình huấn luyện cây GP-Tree, giúp cải thiện tính ổn định và giảm thời gian huấn luyện so với mạng SOM truyền thống. GrSOM có tính linh hoạt cao, cho phép mở rộng mà không cần huấn luyện lại toàn bộ mạng khi có thêm nút lá mới. Mô hình kết hợp này, gọi là SgGP-Tree, được mô tả trong **Hình 3.5.**



Hình 3.5. Mô hình kết hợp SgGP-Tree

Trên cơ sở mạng SOM, cấu trúc mạng GrSOM được định nghĩa như sau:

Định nghĩa 3.2: *Mạng GrSOM*

Mạng GrSOM là một mạng SOM, trong đó đầu vào là các véc-tơ đặc trưng của hình ảnh, ký hiệu là $f = (f_1, f_2, \dots f_m)$, với mỗi véc-tơ f_i có n chiều $f_i = (v_1, v_2, \dots, v_n)$, và giá trị , $f_i \in \{0,1\}$. Tầng đầu ra của mạng chứa các nơ-ron đại diện cho các nút lá trong cây GP-Tree. Các tầng đầu vào và đầu ra được kết nối hoàn toàn thông qua các véc-tơ trọng số $W_i = (w_1, w_2, \dots w_n)$, với $w_i \in \{0,1\}$.

Mạng SgGP-Tree được thiết kế để phân loại dữ liệu đầu vào. Quá trình huấn luyện của nó chủ yếu là điều chỉnh các trọng số. Thay vì khởi tạo trọng số ngẫu nhiên, bộ véc-tơ trọng số đã được huấn luyện từ cây GP-Tree sẽ được sử dụng. Véc-tơ trọng số này được định nghĩa như sau:

Định nghĩa 3.3: *Véc-tơ trọng số*

Gọi w là véc-tơ trọng số của các phần tử dữ liệu ρ tại nút lá. Véc-tơ trọng số này được tính là trung bình của các véc-tơ đặc trưng của các lớp xuất hiện nhiều nhất trong nút lá, với công thức:

$$w = \frac{\sum_{i=1}^{n} f_i}{n}$$

Trong đó, f_i là véc-tơ đặc trưng của các lớp xuất hiện nhiều nhất tại nút lá.

Véc-tơ trọng số được huấn luyện sẽ là tri thức bổ sung để xác định cụm chiến thắng, là cụm tối ưu trên mạng GrSOM. Cụm chiến thắng được xác định như sau:

Định nghĩa 3.4*:* *Cụm chiến thắng*

Gọi f_k là véc-tơ đặc trưng đầu vào của mạng GrSOM, các cụm nút lá L_i, L_j có véc-tơ trọng số như **Định nghĩa 3.3** lần lượt là W_i và W_j. Nếu $sigmoid(\|f_k, W_i\|_2) < sigmoid(\|f_k, W_j\|_2)$, thì L_i là cụm chiến thắng. Cụm chiến thắng L_i kết nối trực tiếp với véc-tơ chiến thắng W_i.

Hàm lỗi của mạng GrSOM được tính bằng:

$$\frac{1}{2}\sum_{i=1}^{N}\left(t_i - f\left(\sum_j w_j . sigmoid\left(\|W_i, W_j\|_2\right)\right)\right)^2$$

Trong đó, t_i là giá trị mong muốn tại nút đầu ra thứ i. Sau khi hoàn tất huấn luyện, ngưỡng θ sẽ được điều chỉnh đủ nhỏ để cập nhật lại các véc-tơ trọng số chiến thắng và các lân cận của chúng.

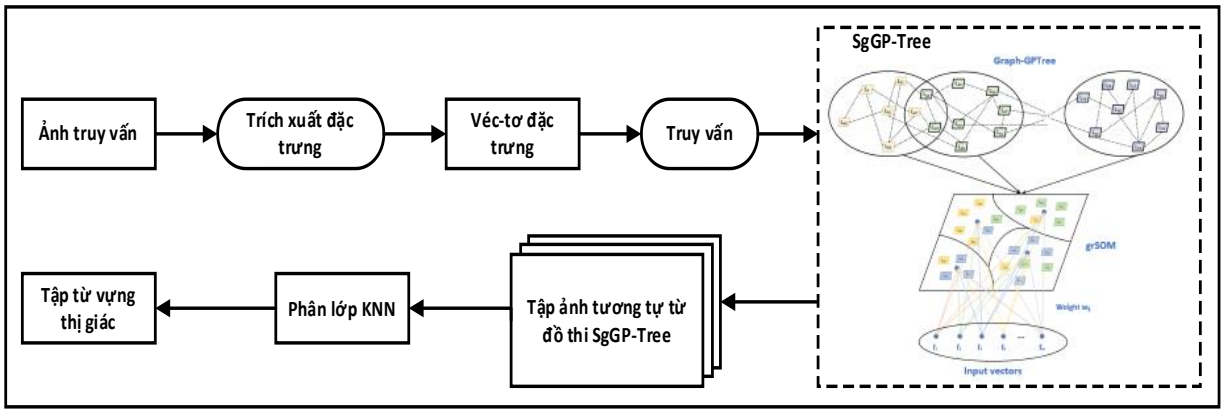
Mạng GrSOM thực hiện phân loại dữ liệu đầu vào qua quá trình huấn luyện ganh đua để chọn cụm chiến thắng, như được mô tả trong **Định nghĩa 3.4**. Hàm kích hoạt *Sigmoid(x)* được sử dụng để xác định cụm chiến thắng, với công thức:

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

Hàm sigmoid(x) này nhận giá trị trong khoảng [0,1], phù hợp với yêu cầu của mạng GrSOM khi đầu ra mong muốn cũng nằm trong khoảng này.

3.3.2. Mô hình tìm kiếm ảnh trên mạng kết hợp SgGP-Tree

Mô hình tìm kiếm ảnh SBIR-SgGP kết hợp cây GP-Tree, đồ thị cụm lân cận Graph-GPTree và mạng SOM, tạo thành cấu trúc SgGP-Tree. Quá trình tiền xử lý bao gồm việc trích xuất và lưu trữ đặc trưng ảnh vào SgGP-Tree. Khi tìm kiếm, SgGP-Tree được sử dụng để tìm các ảnh tương tự và từ vựng thị giác. Khối trích xuất đặc trưng trong mô hình này kế thừa từ các mô hình trước, với cải tiến trong việc kết hợp đồ thị Graph-GPTree và mạng SOM vào cây GP-Tree, tạo nên SgGP-Tree. **Hình 3.6** minh họa mô hình này.



Hình 3.6. Mô trình tìm kiếm ảnh trên SgGP-Tree

Quá trình tìm kiếm ảnh tương tự trên mạng grSOM gồm các bước sau:

(1) Tìm cụm chiến thắng: cụm chiến thắng (Υ) được xác định trên mạng GrSOM dựa trên khoảng cách nhỏ nhất giữ véc-tơ trọng số của cụm đó với véc-tơ đặc trưng f của ảnh tìm kiếm theo độ đo Euclid.

(2) Tìm lân cận của cụm chiến thắng: sau khi xác định cụm chiến thắng, tiếp tục tìm các cụm lân cận xung quanh cụm chiến thắng trên mạng GrSOM do các cụm này có thể chứa các ảnh tương tự với ảnh cần tìm.

(3) Sắp xếp và lựa chọn kết quả: sắp xếp các cụm lân cận dựa trên khoảng cách với cụm chiến thắng. Lựa chọn những cụm có độ tương đồng cao nhất và trích xuất các ảnh từ các cụm này làm kết quả tìm kiếm.

(4) Trả về kết quả: trả về danh sách các ảnh tìm được, được xếp hạng theo mức độ tương đồng với ảnh ban đầu.

Tìm kiếm ảnh trên GrSOM giúp tìm kiếm ảnh trên mạng grSOM một cách hiệu quả bằng cách dựa vào sự tổ chức không gian của các cụm ảnh trên bản đồ tự tổ chức, đảm bảo rằng các ảnh tìm được có độ tương đồng cao với ảnh đầu vào.Thuật toán tìm kiếm ảnh trên mạng GrSOM được mô tả trong **Thuật toán 3.4**

Thuật toán 3.4: Tìm kiếm ảnh trên GrSOM

```
1  Input: Cụm chiến thắng ϒ
2  Output: Tập các hình ảnh tương tự Ω
3  Function: ImRS(ϒ)
4  Begin
5      // Khởi tạo tập hợp ảnh tương tự rỗng
6      Ω = ∅;
7      // Thêm tất cả ảnh từ WinnerCluster vào tập hợp Ω
8       Ω = Ω ∪ ϒ;
9      // Lấy danh sách các cụm lân cận của ϒ
10     ListNeighbor = getNeighbor(ϒ);
11     // Duyệt qua từng cụm lân cận và thêm ảnh của chúng vào Ω
12     Foreach Cluster in ListNeighbor do
13         Ω = Ω ∪ Cluster;
14     EndFor
15     // Trả về tập hợp các ảnh tương tự Ω
16     Return Ω;
17 End.
```

Tính chất 3.4: Độ phức tạp của **Thuật toán 3.4** là $O(h \times log(h) \times k)$.

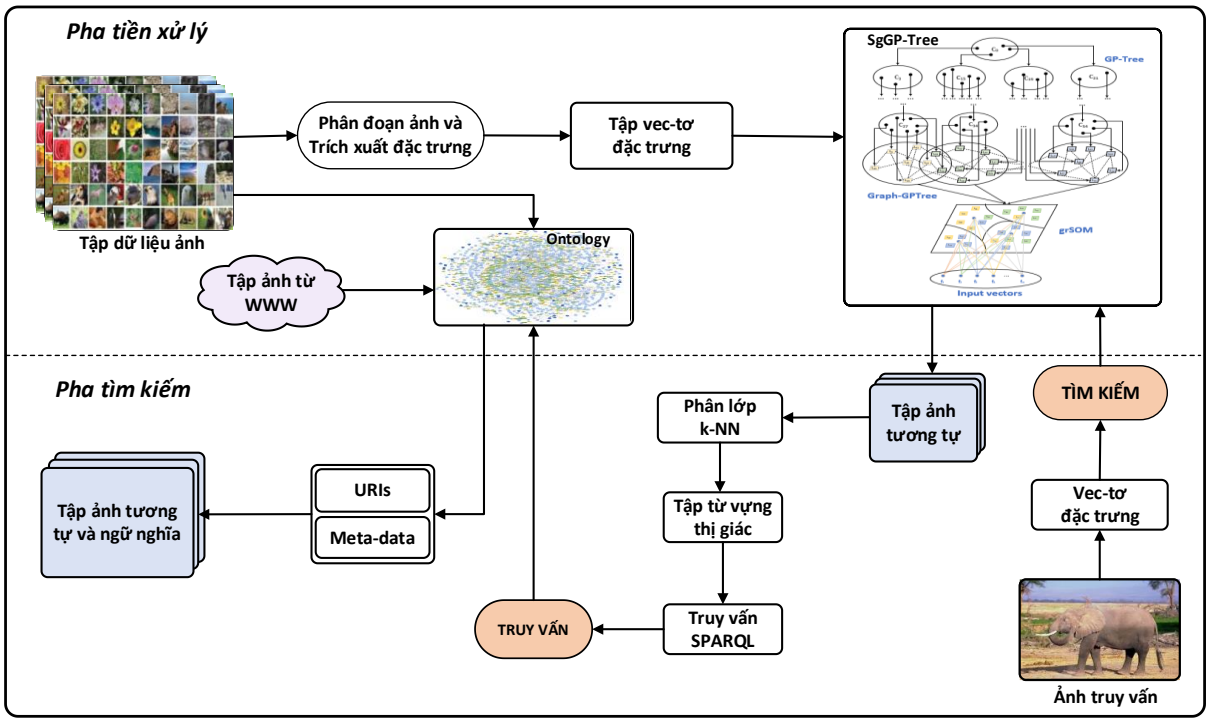
C h ứ n g m i n h:

Trong **Thuật toán 3.4,** độ phức tạp của các bước trong thuật toán như sau: (1) Khởi tạo tập ảnh tương tự Ω có độ phức tạp là $O(1)$; (2) Thêm các ảnh từ Υ vào Ω: việc thêm n ảnh vào tập Ω có độ phức tạp là $O(n)$; (3) Lấy lân cận của Υ: việc lấy danh sách này có độ phức tạp là $O(m)$; (4) Thêm các ảnh từ các cụm lân cận vào Ω: việc thêm tất cả các ảnh từ m cụm lân cận vào Ω có độ phức tạp là $O(m \times p)$. Do đó, tổng độ phức tạp của thuật toán là $O(n + m \times p)$■.

3.4. Hệ tìm kiếm ảnh theo ngữ nghĩa dựa trên ontology

3.4.1. Mô hình tìm kiếm ảnh dựa trên ontology

Hệ thống truy vấn ảnh theo ngữ nghĩa dựa trên ontology, gọi là SBIR-GP, kết hợp cấu trúc học máy SgGP-Tree với ontology. Hệ thống này có hai giai đoạn chính là Tiền xử lý và Truy vấn.



Hình 3.7. Mô hình hệ tìm kiếm SBIR-GP

Hình 3.7 minh họa hệ thống truy vấn ngữ nghĩa dựa trên ontology, gồm hai giai đoạn cụ thể như sau:

Giai đoạn tiền xử lý: (1) Từ tập dữ liệu ảnh, thực hiện kỹ thuật trích chọn và phân đoạn đặc trưng cấp thấp; (2) Tạo tập dữ liệu bao gồm các véc-tơ đặc trưng và phân loại hình ảnh thu được từ quá trình phân đoạn và trích xuất đặc trưng; (3) Từ các mẫu dữ liệu, tạo mô hình kết hợp GP-Tree và Graph-GPTree. Khi các trọng số ban đầu của mạng SOM được khớp với biểu đồ, một tập hợp các véc-tơ trọng số sẽ được huấn luyện; (4) Các mô tả ngữ nghĩa từ bộ sưu tập hình ảnh và WWW được sử dụng để làm phong phú thêm khung ontology đã được xây dựng.

Giai đoạn tìm kiếm ảnh: (1) Hệ thống trích xuất các đặc trưng cấp thấp từ ảnh truy vấn đầu vào; (2) Sử dụng véc-tơ đặc trưng, hệ thống truy vấn SgGP-Tree: tìm kiếm nút lá thích hợp nhất trên GP-Tree, sau đó tìm kiếm tập hợp các nút lá lân cận của nút đó trên Graph-GPTree; đồng thời thực hiện tìm kiếm trên GrSOM để tìm ra cụm ảnh chiến thắng, sau đó lấy các cụm ảnh lân cận của cụm chiến thắng để tìm ra bộ ảnh tương tự tốt nhất; (3) Tìm véc-tơ từ thị giác bằng thuật toán phân loại k-NN trên tập ảnh tương tự. (4) Truy vấn SPARQL được tạo tự động từ véc-tơ từ trực quan; truy vấn được thực thi trên ontology được xây dựng; (5) Kết quả của quá trình truy vấn hình ảnh ngữ nghĩa trên ontology bao gồm siêu dữ liệu, URI, một tập hợp các hình ảnh tương tự và ngữ nghĩa của chúng.

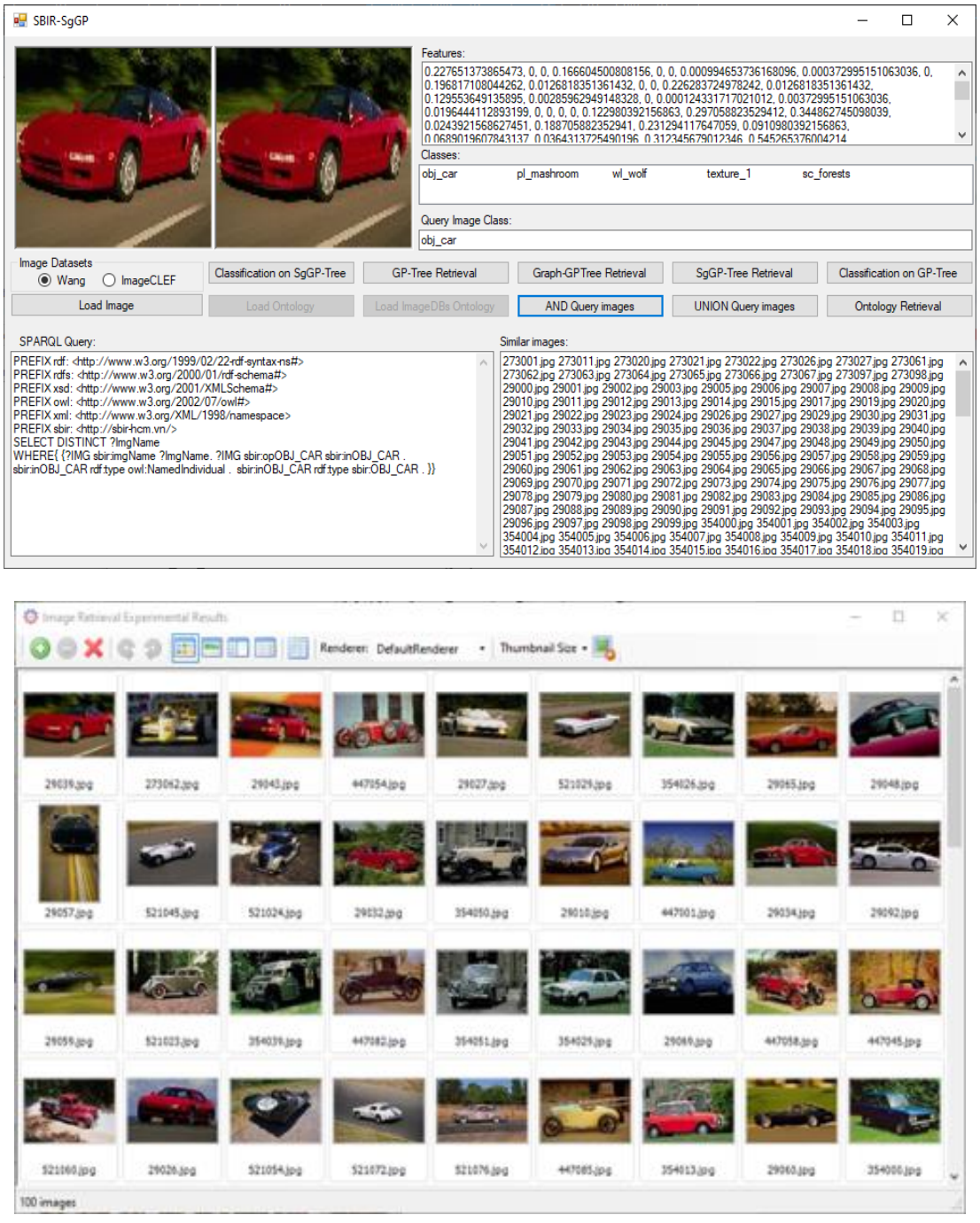
Từ quá trình truy vấn này, cho thấy tập ảnh tương tự cuối cùng được tìm thấy là kết quả giao nhau của các truy vấn trên GP-Tree, Graph-GPTree và GrSOM, nhờ đó đạt được hiệu quả truy vấn hình ảnh tốt nhất trong số các mô hình đề xuất. Đồng thời, việc phân loại trên GrSOM theo cụm chiến thắng sẽ cho kết quả phân loại hình ảnh tốt hơn, dẫn đến kết quả truy vấn chính xác hơn về ontology từ các phân loại này. Kết quả của quá trình truy vấn này là một tập hợp các hình ảnh tương tự, chú thích ngữ nghĩa, mô tả ngữ nghĩa cấp cao và URI/IRI của hình ảnh. Trong kiến trúc tìm kiếm ảnh ngữ nghĩa, ontology đóng một vai trò quan trọng trong việc trích xuất ý nghĩa ngữ nghĩa cấp cao của hình ảnh, bên cạnh việc tổ chức dữ liệu hiệu quả bằng cách sử dụng SgGP-Tree. Kết quả là, một khung ontology được kế thừa và làm phong phú thêm với các bộ dữ liệu bổ sung trong bài viết này để bổ sung thêm các lớp phân cấp (phân loại) và các khái niệm cho các lớp mới.

3.4.2. Thực nghiệm và đánh giá hệ tìm kiếm ảnh SBIR-GP

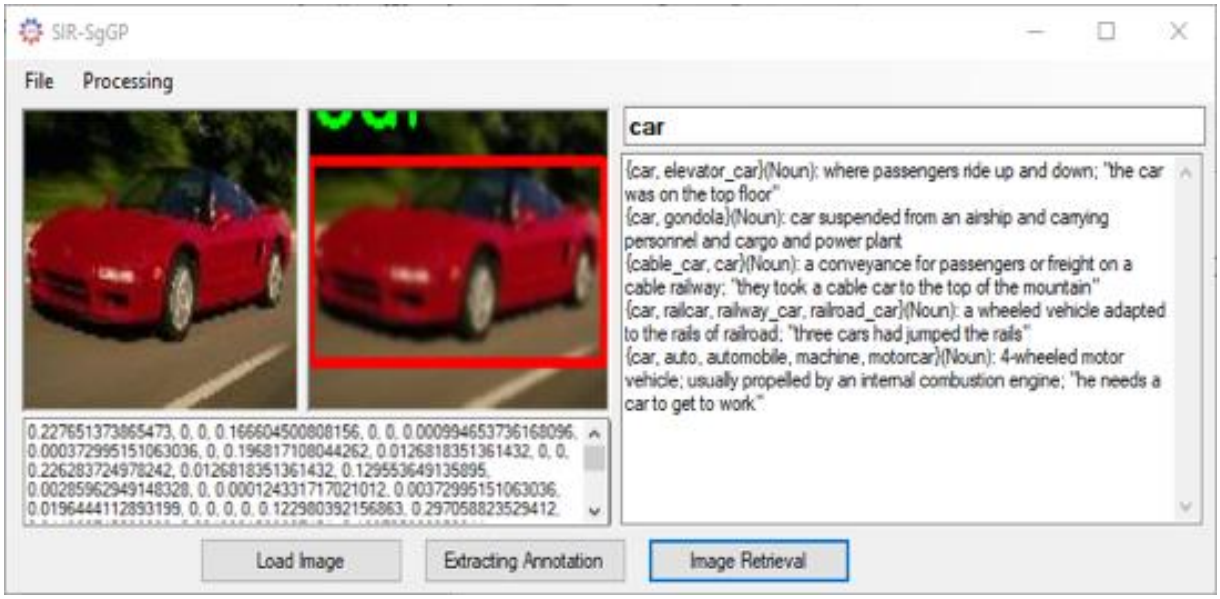
Hệ thống tìm kiếm ảnh SBIR-GP được phát triển để thực hiện các truy vấn ảnh theo ngữ nghĩa bằng cách sử dụng SgGP-Tree và ontology. Khi nhận được một ảnh đầu vào, hệ thống trích xuất các véc-tơ đặc trưng và tìm kiếm các ảnh tương tự dựa trên nội dung bằng SgGP-Tree, từ đó tạo ra một bộ sưu tập ảnh tương tự.

Sau khi có tập ảnh tương tự, hệ thống sẽ phân loại và trích xuất các véc-tơ từ thị giác, đồng thời tạo các truy vấn SPARQL (UNION hoặc AND) để truy vấn ontology. Mỗi

ảnh trong tập hợp sẽ có mô tả ngữ nghĩa, bao gồm siêu dữ liệu và URI. Hệ thống cũng trích xuất các khái niệm ngữ nghĩa từ các từ vựng thị giác thông qua WordNet. **Hình 3.8** là giao diện của hệ OnSBIR với ảnh đầu vào, và kết quả là danh sách các ảnh có ngữ nghĩa tương tự cùng mô tả metadata. **Hình 3.9** minh họa một ví dụ về khái niệm phân lớp từ điển ontology.



Hình 3.8. Một kết quả của hệ tìm kiếm SBIR-GP từ ảnh đầu vào



Hình 3.9. Khái niệm ngữ nghĩa cho lớp

Các bộ dữ liệu hình ảnh được sử dụng cho các thử nghiệm, bao gồm các bộ dữ liệu WANG, MS-COCO và ImageCLEF. Các giá trị hiệu suất trung bình và thời gian tìm kiếm của bộ dữ liệu thử nghiệm được trình bày trong **Bảng 3.1**, **Bảng 3.2** và **Bảng 3.3**

Bảng 3.1. Hiệu suất tìm kiếm ảnh trên bộ dữ liệu ảnh WANG

Phương pháp	Độ chính xác	Độ phủ	Độ dung hòa	Thời gian tìm kiếm trung bình (ms)
GP-Tree	0.6780	0.6840	0.6810	39.75
Graph-GPTree	0.7665	0.6677	0.7137	202.79
SBIR-GP	0.8004	0.7040	0.7491	696.19

Bảng 3.2. Hiệu suất tìm kiếm ảnh trên bộ dữ liệu ảnh ImageCLEF

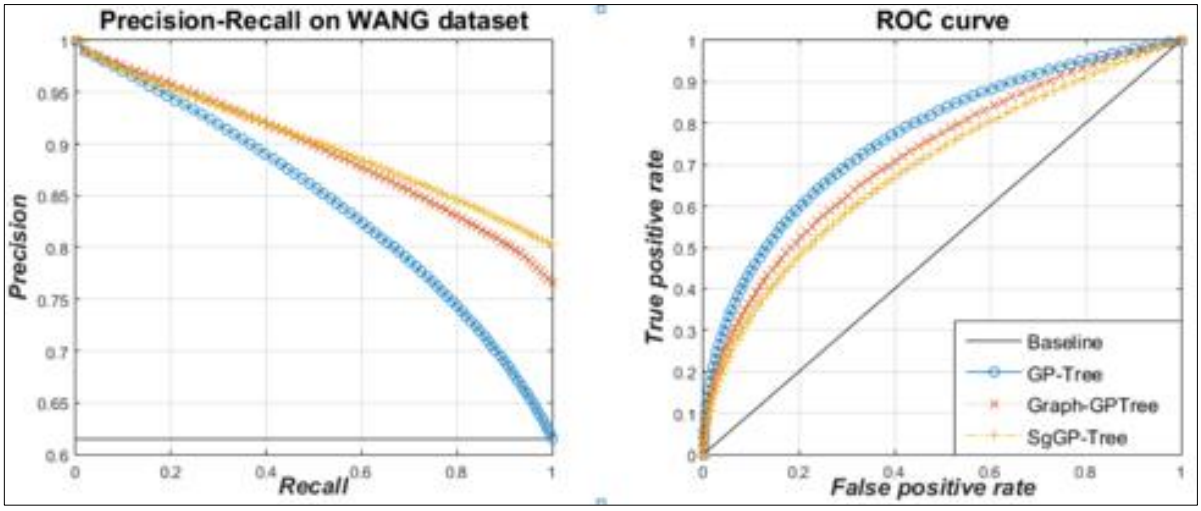
Phương pháp	Độ chính xác	Độ phủ	Độ dung hòa	Thời gian tìm kiếm trung bình (ms)
GP-Tree	0.6802	0.7750	0.7245	44.09
Graph-GPTree	0.8168	0.7637	0.7894	239.29
SBIR-GP	0.8926	0.8764	0.8844	868.51

Bảng 3.3. Hiệu suất tìm kiếm ảnh trên bộ dữ liệu ảnh MS-COCO

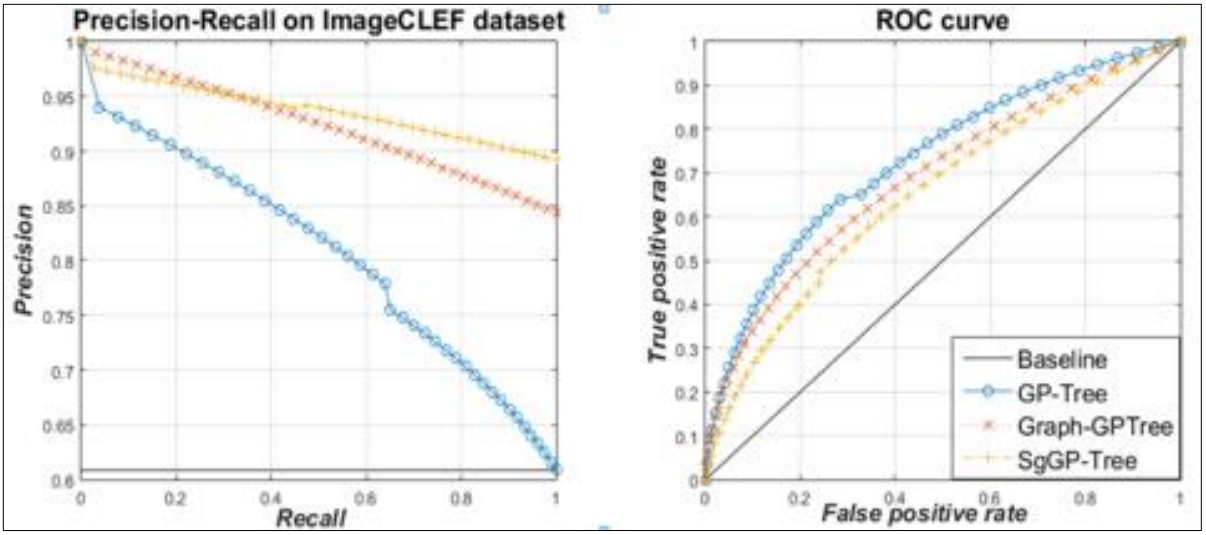
Phương pháp	Độ chính xác	Độ phủ	Độ dung hòa	Thời gian tìm kiếm trung bình (ms)
GP-Tree	0.717	0.724	0.7205	102.32
Graph-GPTree	0.873	0.764	0.815	198.47
SBIR-GP	0.875	0.724	0.783	265.45

Từ các bảng trên, có thể thấy rằng việc cải thiện GP-Tree mang lại hiệu suất tìm kiếm chính xác tốt hơn cho các bộ dữ liệu WANG, ImageCLEF và MS-COCO. Biểu đồ lân cận Graph-GPTree có hiệu suất tốt hơn GP-Tree nhưng thấp hơn SgGP-Tree. Tuy nhiên, thời gian tìm kiếm GP-Tree nhanh hơn Graph-GPTree và SBIR-GP.

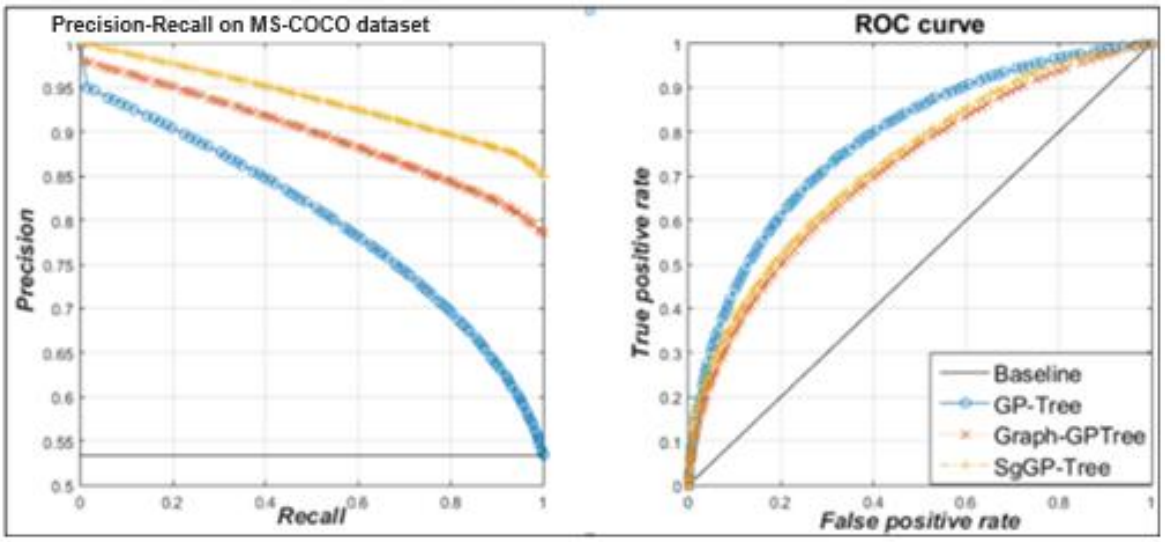
Để đánh giá hiệu quả của hệ thống tìm kiếm, đường cong đặc tính vận hành máy thu (ROC) được sử dụng. Bên cạnh đó, kết hợp độ chính xác và phạm vi bao phủ tạo thành đường cong thu hồi chính xác (PR), cũng có AUC tương tự ROC, với AUC càng lớn thể hiện độ chính xác tốt hơn. Dựa trên dữ liệu thực nghiệm, các đường cong PR và ROC được sử dụng để đánh giá hiệu quả tìm kiếm của hệ thống SBIR-GP **(Hình 3.10, Hình 3.11 và Hình 3.12**).



Hình 3.10. Hiệu suất tìm kiếm ảnh trên GP-Tree, Graph-GPTree và SBIR-GP (SgGP-Tree) trên tập dữ liệu ảnh WANG.



Hình 3.11. Hiệu suất tìm kiếm ảnh trên GP-Tree, Graph-GPTree và SBIR-GP (SgGP-Tree) trên tập dữ liệu ảnh ImageCLEF.



Hình 3.12. Hiệu suất tìm kiếm ảnh trên GP-Tree, Graph-GPTree và SBIR-GP (SgGP-Tree) trên tập dữ liệu ảnh MS-COCO.

Trong đồ thị PR, mỗi đường cong đại diện cho một thư mục hình ảnh trong từng tập dữ liệu. Đường cong Precision-Recall cho thấy SBIR-GP (SgGP-Tree) có diện tích dưới đường cong lớn nhất, tiếp theo là Graph-GPTree và thấp nhất là GP-Tree, cho thấy các cải tiến đã nâng cao độ chính xác. Trong đồ thị ROC, đường chéo cơ sở phân chia không gian thành hai phần, với các điểm trên đường chéo biểu thị kết quả phân loại chính xác, và dưới đường chéo là kết quả phân loại sai. Đường cong ROC của hệ thống cho thấy kết quả phân loại hình ảnh tốt, với Graph-GPTree đạt hiệu suất cao hơn GP-Tree nhưng không bằng SBIR-GP (SgGP-Tree). Hiệu suất tìm kiếm ảnh trên bộ dữ liệu WANG, ImageCLEF và MS-COCO chứng minh hiệu quả của phương pháp cải tiến. Để đánh giá chính xác và hiệu quả của hệ thống, kết quả độ chính xác trung bình của các phương pháp tìm kiếm trên các bộ dữ liệu này được so sánh trong **Bảng 3.4, 3.5 và 3.6**.

Bảng 3.4. So sánh các phương pháp tìm kiếm ảnh trên bộ dữ liệu ảnh WANG

Phương pháp	Độ chính xác trung bình
K. Kanwal và cộng sự, 2020 [66]	0.5067
H. Zeng và cộng sự, 2021 [77]	0.6600

O. Sikha và K. Soman, 2021 [78]	0.8030
S. Dhingra và P. Bansal, 2021 [67]	0.6000
A. Ouni và cộng sự, 2022 [79]	0.7800
Graph-GPTree	**0.7665**
SBIR-GP (SgGP-Tree)	**0.8004**

Bảng 3.5. So sánh các phương pháp tìm kiếm ảnh trên bộ dữ liệu ảnh ImageCLEF

Phương pháp	**Độ chính xác trung bình**
A. Yang và cộng sự, 2019 [80]	0.8030
Y. Qiang và cộng sự, 2020 [68]	0.6670
X. Yue và cộng sự, 2021 [69]	0.7140
N. T. U. Nhi và cộng sự, 2022 [70]	0.6510
X. Wang và cộng sự, 2023 [71]	0.7270
Graph-GPTree	**0.8168**
SgGP-Tree (SBIR-GP)	**0.8926**

Bảng 3.6. So sánh các phương pháp tìm kiếm ảnh trên bộ dữ liệu ảnh MS-COCO

Phương pháp	**Độ chính xác trung bình**
Y. Cao và cộng sự, 2018 [81]	0.8576
Y. Xie và cộng sự, 2020 [82]	0.8628
Wen Gu và cộng sự, 2019 [83]	0.8350
Graph-GPTree	**0.8730**
SgGP-Tree (SBIR-GP)	**0.8753**

Bảng 3.4 so sánh kết quả phương pháp đề xuất với các nghiên cứu trước trên bộ dữ liệu WANG (10.800 ảnh). Graph-GPTree đạt độ chính xác cao hơn một số phương pháp khác, nhưng vẫn thấp hơn kết quả của O. Sikha và K. Soman [78] cùng A. Ouni [79]. SgGP-Tree kết hợp mạng SOM cải thiện độ chính xác, vượt trội hơn [79], nhưng vẫn chưa đạt được kết quả của [78]. Mặc dù có sự cải thiện không lớn, các phương pháp đề xuất vẫn chứng tỏ hiệu quả đối với bộ dữ liệu WANG.

Bảng 3.5 và **3.6** cho bộ ảnh ImageCLEF cho thấy Graph-GPTree đạt hiệu quả tìm kiếm ảnh vượt trội so với các phương pháp khác. Vì bộ ảnh ImageCLEF có nhiều đối tượng và phân tán cao, việc tìm kiếm chính xác là một thử thách. Tuy nhiên, nhờ sử dụng đồ thị cụm lân cận Graph-GPTree, kết quả tìm kiếm được cải thiện rõ rệt. SgGP-Tree, phát triển từ Graph-GPTree, giúp nâng cao độ chính xác tìm kiếm, chứng minh hiệu quả của các cải tiến đối với bộ ảnh ImageCLEF và MS-COCO.

Kết quả trên cho thấy phương pháp đề xuất đạt độ chính xác cao hơn các phương pháp tìm kiếm khác trên cùng bộ dữ liệu. Phương pháp này có khả năng trích xuất hiệu quả đặc trưng và phân biệt các chi tiết của đối tượng trong ảnh, giúp giải quyết bài toán truy vấn và phân tích ngữ nghĩa đối với ảnh đơn và đa đối tượng.

3.5. Tiểu kết chương

Trong chương này, các phương pháp nâng cao hiệu suất tra cứu ảnh trên GP-Tree được đề xuất. Đầu tiên, một mô hình kết hợp giữa biểu đồ lân cận và GP-Tree, được gọi là Graph-GPTree, đã được đề xuất nhằm khắc phục hạn chế của GP-Tree khi các phần tử tương tự bị phân tán qua các nhánh khác nhau trong quá trình phân tách nút. Bằng cách sử dụng biểu đồ lân cận, Graph-GPTree kết nối các phần tử tương tự nằm ở các nhánh khác nhau, giúp tăng cường khả năng tìm kiếm chính xác và hiệu quả hơn.

Tiếp theo, một mô hình tiên tiến hơn được phát triển bằng cách kết hợp giữa GrSOM và Graph-GPTree, được gọi là SgGP-Tree. Mô hình này bổ sung các tiêu chí nhằm chọn ra các nút lá chiến thắng dựa trên sự tương đồng, từ đó cải thiện quá trình phân cụm và nâng cao độ chính xác trong việc tìm kiếm ảnh. SgGP-Tree mang lại sự tối ưu trong việc phân chia các nút lá và cải thiện rõ rệt chất lượng tra cứu hình ảnh.

Để kiểm tra tính hiệu quả của các mô hình đề xuất, các thực nghiệm đã được tiến hành trên ba bộ dữ liệu ảnh nổi tiếng là WANG, ImageCLEF, và MS-COCO. Hệ thống SBIR-GP, dựa trên mô hình SgGP-Tree, đã chứng minh độ chính xác vượt trội so với các đề xuất trước đây của tác giả luận án. Đặc biệt, hiệu suất của hệ thống SBIR-GP được so sánh trực tiếp với các phương pháp khác trên cùng tập dữ liệu hình ảnh, từ đó đánh giá khả năng của các mô hình, phương pháp và thuật toán được đề xuất trong chương.

Kết quả so sánh cho thấy hệ thống tra cứu SBIR-GP không chỉ đạt được độ chính xác cao hơn các nghiên cứu trước mà còn vượt trội so với các phương pháp tra cứu ảnh khác trên cùng tập dữ liệu thực nghiệm. Điều này chứng minh rằng các phương pháp và mô hình được đề xuất trong chương là hiệu quả, mang tính khả thi và có tiềm năng ứng dụng thực tiễn cao trong bài toán tìm kiếm ảnh theo nội dung và ngữ nghĩa.

KẾT LUẬN

1. Đóng góp của luận án

Trong luận án này, các phương pháp tìm kiếm ảnh dựa trên ngữ nghĩa đã được đề xuất và phát triển qua việc phân tích sâu rộng các nghiên cứu liên quan, nhằm xây dựng những mô hình tìm kiếm ảnh có hiệu suất cao. Luận án tập trung vào việc kết hợp các kỹ thuật phân cụm và ontology, với ba đóng góp chính:

(1) Xây dựng cấu trúc GP-Tree để lưu trữ và lập chỉ mục dữ liệu ảnh: GP-Tree dựa trên phân cụm phân cấp, cho phép lưu trữ hiệu quả dữ liệu ảnh lớn thông qua các véc-tơ đặc trưng cấp thấp. Sự kết nối giữa đặc trưng ảnh và từ vựng ngữ nghĩa giúp giảm thiểu kích thước dữ liệu, đồng thời tăng tốc độ và độ chính xác của quá trình tìm kiếm.

(2) Hệ thống tìm kiếm ảnh theo ngữ nghĩa GP-SBIR dựa trên GP-Tree: GP-SBIR kết hợp GP-Tree với ontology, hoạt động qua hai giai đoạn: tiền xử lý và tìm kiếm. Giai đoạn tiền xử lý xây dựng GP-Tree và khung ontology bán tự động dựa trên RDF, trong khi giai đoạn tìm kiếm thực hiện tìm kiếm ảnh tương tự và truy vấn ngữ nghĩa thông qua SPARQL. Độ chính xác trên các bộ dữ liệu WANG, ImageCLEF, và MS-COCO đạt lần lượt 0.6780, 0.6802, và 0.7170.

(3) Phát triển GP-Tree với hai mô hình mới là Graph-GPTree và SgGP-Tree: Graph-GPTree cải thiện độ chính xác tìm kiếm bằng cách liên kết các phần tử tương tự nằm trên các nhánh khác nhau của cây. Trong khi đó, SgGP-Tree kết hợp với SOM để tối ưu hóa phân cụm và chọn các nút lá hiệu quả hơn, từ đó nâng cao hiệu suất tìm kiếm. Thử nghiệm trên các bộ dữ liệu WANG, ImageCLEF, và MS-COCO cho thấy độ chính xác của Graph-GPTree lần lượt đạt 0.7665, 0.8168, và 0.8730, trong khi SgGP-Tree đạt 0.8004, 0.8926, và 0.8753. Những kết quả này khẳng định rằng các mô hình cải tiến GP-Tree mang lại hiệu quả cao trong việc tìm kiếm ảnh.

Luận án đã tiến hành thử nghiệm trên các tập dữ liệu nổi tiếng như WANG, ImageCLEF, và MS-COCO, và thu được kết quả tốt hơn về độ chính xác so với các phương pháp tìm

kiếm hiện đại khác. Những kết quả này không chỉ nâng cao hiệu quả tìm kiếm ảnh dựa trên ngữ nghĩa mà còn mở ra tiềm năng phát triển trong lĩnh vực này.

2. Hướng phát triển

Mặc dù đã đạt được những kết quả đáng kể, luận án vẫn có một số hướng phát triển mở rộng như sau:

(1) So sánh với các phương pháp hiện đại: Nghiên cứu các phương pháp tìm kiếm ảnh sử dụng mạng nơ-ron sâu (DNN), CNN, R-CNN, GCN,... để so sánh hiệu quả với các phương pháp đã đề xuất trong luận án.

(2) Ứng dụng thực tế: Phát triển ứng dụng tìm kiếm ảnh ngữ nghĩa trong các lĩnh vực như xác định địa điểm du lịch từ ảnh, chẩn đoán bệnh qua ảnh y khoa, phân loại đá thổ nhưỡng, và tìm kiếm ảnh trên mạng xã hội.

(3) Mở rộng ontology ngữ nghĩa: Phát triển thêm các mối quan hệ ngữ nghĩa giữa các đối tượng và hành động trong ảnh để nâng cao độ chính xác trong việc hiểu và tìm kiếm ảnh.

(4) Phát triển ontology tiếng Việt: Xây dựng ontology cho tìm kiếm ảnh ngữ nghĩa bằng tiếng Việt, mở rộng ứng dụng cho người dùng trong nước.

DANH MỤC CÁC CÔNG TRÌNH CÔNG BỐ

[CT1] **N. M. Hai**, T. V. Lang, and V. T. Thanh, "Semantic-Based Image Retrieval Using Hierarchical Clustering and Neighbor Graph," in World Conference on Information Systems and Technologies, 2022, pp. 34-44: Springer, DOI: https://doi.org/10.1007/978-3-031-04829-6_4 (**Scopus, Q4**)

[CT2] **N. M. Hai**, V. T. Thanh, and T. V. Lang, "A method for semantic-based image retrieval using hierarchical clustering tree and graph," Telkomnika, vol. 20, no. 5, pp. 1026-1033, 2022, DOI: http://doi.org/10.12928/telkomnika.v20i5.24086 (**Scopus, Q3**)

[CT3] **N. M. Hai**, T. V. Lang and T. The Van, "Improving the Efficiency of Semantic Image Retrieval Using a Combined Graph and SOM Model," in IEEE Access, vol. 11, pp. 140646-140659, 2023, doi: https://doi.org/10.1109/ACCESS.2023.3333678 (**SCIE, Q1**)

[CT4] **N. M. Hai**, V. T. Thanh, and T. V. Lang, "The improvements of semantic-based image retrieval using hierarchical clustering tree," in Proceedings of the 13th National Conference on Fundamental and Applied Information Technology Research (FAIR'2020), 2020, pp. 557-570: Natural Science and Technology Publishing House, DOI: https://doi.org/10.15625/vap.2020.00213

[CT5] **N. M. Hai**, V. T. Thanh, and T. V. Lang, "A method of semantic-based image retrieval using graph cut", Journal of Computer Science and Cybernetics, vol. 38, no. 2, pp. 193-212, 2022, DOI: https://doi.org/10.15625/1813-9663/38/2/16786

[CT6] **Nguyễn Minh Hải**, Trần Văn Lăng, Văn Thế Thành, "Một tiếp cận tìm kiếm ảnh theo ngữ nghĩa dựa trên mạng nơ-ron tích chập và ontology", Tạp chí khoa học Trường ĐH Sư phạm TP. HCM, 2022. tr. 48-59. DOI: https://doi.org/10.54607/hcmue.js.19.3.3272(2022)

TÀI LIỆU THAM KHẢO

[1] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, pp. 1-60, 2008.

[2] C. H. Leung and Y. Li, "Semantic Enrichment for Automatic Image Retrieval," in *Semantic Multimedia Analysis and Processing*: CRC Press, 2017, pp. 111-132.

[3] S. Jain, K. Pulaparthi, and C. Fulara, "Content based image retrieval," *Int. J. Adv. Eng. Glob. Technol*, vol. 3, pp. 1251-1258, 2015.

[4] K. D. Martin *et al.*, "Data privacy in retail," *Journal of Retailing*, vol. 96, no. 4, pp. 474-489, 2020.

[5] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. J. J. o. b. d. Muharemagic, "Deep learning applications and challenges in big data analytics," vol. 2, no. 1, pp. 1-21, 2015.

[6] L. Wang and L. Khan, "Automatic image annotation and retrieval using weighted feature selection," *Multimedia Tools Applications*, vol. 29, pp. 55-71, 2006.

[7] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. G. Snoek, and A. D. Bimbo, "Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval," *ACM Computing Surveys*, vol. 49, no. 1, pp. 1-39, 2016.

[8] B. d. B. Pereira, C. R. Rao, R. L. Oliveira, and E. M. do Nascimento, "Combining unsupervised and supervised neural networks in cluster analysis of gamma-ray burst," *Journal of Data Science*, vol. 8, pp. 327-338, 2010.

[9] J.-M. Guo and H. Prasetyo, "Content-based image retrieval using features extracted from halftoning-based block truncation coding," *IEEE Transactions on image processing*, vol. 24, no. 3, pp. 1010-1024, 2014.

[10] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern recognition*, vol. 40, no. 1, pp. 262-282, 2007.

[11] Z. Xia *et al.*, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE transactions on information forensics*, vol. 11, no. 11, pp. 2594-2608, 2016.

[12] W. Database. (2021). *Wang Database*. Available: http://wang.ist.psu.edu/docs/related/,

[13] MS-COCO. (2017). *Dataset MS-COCO 2017*. Available: https://www.kaggle.com/datasets/awsaf49/coco-2017-dataset?resource=download

[14] L. M. Thanh and e. al, "Image retrieval system based on EMD similarity measure and S-tree," (in V), *Intelligent Technologies and Engineering Systems, Springer, New York, NY*, pp. 139-146, 2013.

[15] H. H. Wang, D. Mohamad, and N. A. Ismail, "Approaches, challenges and future direction of image retrieval," *arXiv preprint arXiv*, 2010.

[16] Y. Rui, T. S. Huang, and S.-F. Chang, "Image retrieval: Current techniques, promising directions, and open issues," *Journal of visual communication image representation*, vol. 10, no. 1, pp. 39-62, 1999.

[17] M. Singha and K. Hemachandran, "Content based image retrieval using color and texture," *Signal Image Processing,* vol. 3, no. 1, p. 39, 2012.
[18] O. Allani, H. B. Zghal, N. Mellouli, H. Akdag, and Applications, "Pattern graph-based image retrieval system combining semantic and visual features," *Multimedia Tools Applications,* vol. 76, no. 19, pp. 20287-20316, 2017.
[19] V. P. Singh, R. Srivastava, and B. Engineering, "Automated and effective content-based mammogram retrieval using wavelet based CS-LBP feature and self-organizing map," *Biocybernetics,* vol. 38, no. 1, pp. 90-105, 2018.
[20] L. Piras and G. J. I. F. Giacinto, "Information fusion in content based image retrieval: A comprehensive overview," vol. 37, pp. 50-60, 2017.
[21] S. Bruch, *Foundations of Vector Retrieval*. Springer, 2024.
[22] C.-M. Lo, "Multimedia information retrieval using content-based image retrieval and context link for Chinese cultural artifacts," *Library Hi Tech,* 2024.
[23] Z. Liu and J. Bonar, "Differential Shape Optimization with Image Representation for Photonic Design," *arXiv preprint arXiv:.13074,* 2024.
[24] L. R. Nair, K. Subramaniam, G. PrasannaVenkatesan, P. Baskar, and T. Jayasankar, "Essentiality for bridging the gap between low and semantic level features in image retrieval systems: an overview," *Journal of Ambient Intelligence Humanized Computing,* vol. 12, pp. 5917-5929, 2021.
[25] W. Li, L. Duan, D. Xu, and I. W.-H. Tsang, "Text-based image retrieval using progressive multi-instance learning," in *2011 international conference on computer vision*, 2011, pp. 2049-2055: IEEE.
[26] D. Srivastava, S. S. Singh, B. Rajitha, M. Verma, M. Kaur, and H.-N. J. I. A. Lee, "Content-based Image Retrieval: A Survey on Local and Global Features Selection, Extraction, Representation and Evaluation Parameters," 2023.
[27] C. C. L. Wenyin and H. Zhang, "Image retrieval based on region shape similarity."
[28] M. Garg and G. Dhiman, "A novel content-based image retrieval approach for classification using GLCM features and texture fused LBP variants," *Neural Computing Applications,* vol. 33, no. 4, pp. 1311-1328, 2021.
[29] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: an experimental comparison," *Information retrieval,* vol. 11, pp. 77-107, 2008.
[30] M. Azimi Hemat, "Fuzzy Content-Based Image Retrieval Speed-up Using the Multi-Agent Platform," *AUT Journal of Modeling Simulation,* vol. 54, no. 1, pp. 3-18, 2022.
[31] E. Kiamansouri, H. Barati, and A. Barati, "A two-level clustering based on fuzzy logic and content-based routing method in the internet of things," *Peer-to-Peer Networking Applications,* vol. 15, no. 4, pp. 2142-2159, 2022.
[32] M. K. Yusof, "Effectiveness of Dominant Color Descriptor Technique in Medical Image Retrieval Application," *World Academy of Science, Engineering Technology, International Science Index,* 2010.
[33] M. K. Alsmadi, "Content-based image retrieval using color, shape and texture descriptors and features," *Arabian Journal for Science Engineering,* vol. 45, no. 4, pp. 3317-3330, 2020.

[34] Y. Xu, Y. Bin, J. Wei, Y. Yang, G. Wang, and H. T. Shen, "Multi-modal transformer with global-local alignment for composed query image retrieval," *IEEE Transactions on Multimedia*, vol. 25, pp. 8346-8357, 2023.

[35] E. Winarno, K. Nugroho, and P. W. Adi, "Combined interleaved pattern to improve confusion-diffusion image encryption based on hyperchaotic system," *IEEE Access*, vol. 11, pp. 69005-69021, 2023.

[36] Q. Zhang, Z. Lei, Z. Zhang, and S. Z. Li, "Context-aware attention network for image-text retrieval," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3536-3545.

[37] A. H. Sơn, "Tra cứu ảnh dựa vào nội dung với học biểu diễn và giảm chiều dữ liệu," Luận án tiến sĩ, Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, 2023.

[38] R. S. Wu and W. H. Hsu, "A Semantic Image Retrieval Frame work based on Ontology and Naïve Bayesian Inference," *International Journal of Multimedia Technology*, vol. 2, no. 2, pp. 36-43, 2012.

[39] X. Wang, S. Qiu, K. Liu, and X. Tang, "Web image re-ranking usingquery-specific semantic signatures," *IEEE transactions on pattern analysis machine intelligence*, vol. 36, no. 4, pp. 810-823, 2013.

[40] Đ. T. T. Quỳnh, "Nâng cao độ chính xác tra cứu ảnh dựa vào nội dung sử dụng kỹ thuật điều chỉnh trọng số hàm khoảng cách," Luận án tiến sĩ, Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, 2019.

[41] W. Hu, Y. Sheng, X. Zhu, and M. Computing, "A Semantic Image Retrieval Method Based on Interest Selection," *Wireless Communications*, vol. 2022, 2022.

[42] Y. Shi, X. Liu, Y. Wei, Z. Wu, and W. Zuo, "Retrieval-based Spatially Adaptive Normalization for Semantic Image Synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11224-11233.

[43] U. Manzoor, M. A. Balubaid, B. Zafar, H. Umar, and M. S. Khan, "Semantic image retrieval: An ontology based approach," *International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 4, pp. 1-8, 2015.

[44] S. Chaudhury, A. Mallik, and H. Ghosh, *Multimedia ontology: representation and applications*. CRC Press, 2015.

[45] S. A. Fadzli and R. Setchi, "Semantic approach to image retrieval using statistical models based on a lexical ontology," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2010, pp. 240-250: Springer.

[46] N. Ruan, N. Huang, and W. Hong, "Semantic-based image retrieval in remote sensing archive: An ontology approach," in *2006 IEEE International Symposium on Geoscience and Remote Sensing*, 2006, pp. 2903-2906: IEEE.

[47] N. Magesh and P. Thangaraj, "Semantic image retrieval based on ontology and SPARQL query," in *International Conference on Advanced Computer Technology (ICACT)*, 2011.

[48] Y. Liu, Y. Huang, S. Zhang, D. Zhang, and N. Ling, "Integrating object ontology and region semantic template for crime scene investigation image retrieval," in

2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA), 2017, pp. 149-153: IEEE.

[49] M. S. Sulaiman, S. Nordin, and N. Jamil, "An object properties filter for multi-modality ontology semantic image retrieval," *Journal of Information Communication Technology,* vol. 16, no. 1, pp. 1-19, 2017.

[50] A. B. Spanier, D. Cohen, and L. Joskowicz, "A new method for the automatic retrieval of medical cases based on the RadLex ontology," *International journal of computer assisted radiology and surgery,* vol. 12, no. 3, pp. 471-484, 2017.

[51] L. M. P. T. C. An, "Tìm kiếm ảnh theo nội dung và ngữ nghĩa," *Tap chi Khoa hoc Truong ĐH Can Tho,* no. CNTT, pp. 58-64, 2017.

[52] D. K. McClish, "Analyzing a portion of the ROC curve," *Medical decision making,* vol. 9, no. 3, pp. 190-195, 1989.

[53] K. Boyd, K. H. Eng, and C. D. Page, "Area under the precision-recall curve: point estimates and confidence intervals," in *Joint European conference on machine learning and knowledge discovery in databases*, 2013, pp. 451-466: Springer.

[54] A. Gordoa, J. A. Rodriguez-Serrano, F. Perronnin, and E. Valveny, "Leveraging category-level labels for instance-level image retrieval," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3045-3052: IEEE.

[55] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, 2016, pp. 241-257: Springer.

[56] C. H. Song, J. Yoon, S. Choi, and Y. Avrithis, "Boosting vision transformers for image retrieval," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 107-117.

[57] M. Wang, W. Zhou, Q. Tian, and H. Li, "Deep graph convolutional quantization networks for image retrieval," *IEEE Transactions on Multimedia,* vol. 25, pp. 2164-2175, 2022.

[58] S. Gkelios, A. Sophokleous, S. Plakias, Y. Boutalis, and S. A. Chatzichristofis, "Deep convolutional features for image retrieval," *Expert Systems with Applications*, vol. 177, p. 114940, 2021.

[59] I. M. Hameed, S. H. Abdulhussain, and B. M. Mahmmod, "Content-based image retrieval: A review of recent trends," *Cogent Engineering*, vol. 8, no. 1, p. 1927469, 2021.

[60] N. T. U. Nhi, & Le, T. M., "A Model of Semantic-Based Image Retrieval Using C-Tree and Neighbor Graph," *International Journal on Semantic Web and Information Systems (IJSWIS)*, pp. 1-23, 2022.

[61] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961-2969.

[62] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis machine intelligence,* vol. 35, no. 8, pp. 1798-1828, 2013.

[63] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," (in E), *In Proceedings of the IEEE conference on computer vision and pattern recognition* pp. 6154-6162, 2018.

[64] M. I. T. Bella and A. Vasuki, "An efficient image retrieval framework using fused information feature," (in E), *Computers & Electrical Engineering*, vol. 75, pp. 46-60, 2019.

[65] P. Chhabra, N. K. Garg, and M. Kumar, "Content-based image retrieval system using ORB and SIFT features," (in E), *Neural Computing and Applications*, pp. 2725-2733, 2020.

[66] K. Kanwal, K. T. Ahmad, R. Khan, A. T. Abbasi, and J. Li, "Deep learning using symmetry, FAST scores, shape-based filtering and spatial mapping integrated with CNN for large scale image retrieval," *Symmetry*, vol. 12, no. 4, p. 612, 2020.

[67] S. Dhingra and P. Bansal, "Relative examination of texture feature extraction techniques in image retrieval systems by employing neural network: an experimental review," in *Proceedings of International Conference on Artificial Intelligence and Applications: ICAIA 2020*, 2021, pp. 337-349: Springer.

[68] Y. Qiang, C. Sheng, and D. Yin, "Method of tire pattern image retrieval based on wavelet transform and Siamese network," in *Proceedings of the 2020 International Conference on Aviation Safety and Information Technology*, 2020, pp. 587-592.

[69] X. Yue *et al.*, "Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13834-13844.

[70] N. T. U. Nhi and T. M. Le, "A model of semantic-based image retrieval using C-tree and neighbor graph," *International Journal on Semantic Web Information Systems*, vol. 18, no. 1, pp. 1-23, 2022.

[71] X. Wang, D. Peng, M. Yan, and P. Hu, "Correspondence-free domain alignment for unsupervised cross-domain image retrieval," *arXiv preprint arXiv:.06081*, 2023.

[72] J. Wang, et al., "Cnn-rnn: A unified framework for multi-label image classification," (in E), *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[73] A. Wang, Y. Wang, and Y. Chen, "Hyperspectral image classification based on convolutional neural network and random forest," (in E), *Remote sensing letters*, vol. 10, no. 11, pp. 1086-1094, 2019.

[74] S. Wen *et al.*, "Multilabel image classification via feature/label co-projection," (in E), *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, 2020.

[75] Z. Zhang, A. Peng, and H. Li, "Instance-weighted central similarity for multi-label image retrieval," (in E), *arXiv preprint arXiv*, vol. 2108.05274, 2021.

[76] C. S. Wickramasinghe, K. Amarasinghe, and M. Manic, "Parallalizable deep self-organizing maps for image classification," in *2017 IEEE symposium series on computational intelligence (SSCI)*, 2017, pp. 1-7: IEEE.

[77] G.-H. Liu and J.-Y. Yang, "Deep-seated features histogram: a novel image retrieval method," *Pattern Recognition*, vol. 116, p. 107926, 2021.

[78] O. Sikha and K. Soman, "Dynamic Mode Decomposition based salient edge/region features for content based image retrieval," *Multim. Tools Appl.*, vol. 80, no. 10, pp. 15937-15958, 2021.

[79] A. Ouni, E. Royer, M. Chevaldonné, and M. Dhome, "Leveraging semantic segmentation for hybrid image retrieval methods," *Neural Computing Applications*, vol. 34, no. 24, pp. 21519-21537, 2022.

[80] A. Yang, X. Yang, W. Wu, H. Liu, and Y. J. I. a. Zhuansun, "Research on feature extraction of tumor image based on convolutional neural network," vol. 7, pp. 24204-24213, 2019.

[81] Y. Cao, M. Long, B. Liu, and J. Wang, "Deep cauchy hashing for hamming space retrieval," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1229-1237.

[82] Y. Xie, Y. Liu, Y. Wang, L. Gao, P. Wang, and K. Zhou, "Label-Attended Hashing for Multi-Label Image Retrieval," in *IJCAI*, 2020, pp. 955-962.

[83] W. Gu, X. Gu, J. Gu, B. Li, Z. Xiong, and W. Wang, "Adversary guided asymmetric hashing for cross-modal retrieval," in *Proceedings of the 2019 on international conference on multimedia retrieval*, 2019, pp. 159-167.