

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



MAI TIẾN DŨNG

**PHÁT TRIỂN MỘT SỐ PHƯƠNG PHÁP
PHÂN LOẠI ẢNH VỚI SỐ LƯỢNG LỚP LỚN**

Chuyên ngành: KHOA HỌC MÁY TÍNH

Mã số: 62.48.01.01

TÓM TẮT LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

TP. HỒ CHÍ MINH – 2017

Công trình được hoàn thành tại:

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN -
ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

Người hướng dẫn khoa học: GS.TSKH. HOÀNG VĂN KIỂM

Phản biện 1:

Phản biện 2:

Phản biện 3:

Luận án sẽ được bảo vệ trước

Hội đồng chấm luận án cấp Trường tại :

.....

.....

vào lúc giờ ngày tháng năm .

Có thể tìm hiểu luận án tại:

- Thư viện Quốc gia Việt nam.
- Thư viện Trường Đại học Công nghệ Thông tin – Đại học Quốc gia
Tp. Hồ Chí Minh.

Mục lục

Mục lục	1
Chương 1. TỔNG QUAN	4
1.1 Giới thiệu tóm tắt về công trình nghiên cứu	4
1.2 Động lực nghiên cứu	6
1.3 Mục đích, đối tượng và phạm vi nghiên cứu	6
1.4 Ý nghĩa khoa học và thực tiễn của đề tài	7
1.5 Bố cục luận án	8
Chương 2. CƠ SỞ LÝ THUYẾT	9
2.1 Giới thiệu bài toán	9
2.1.1 Bài toán đa phân lớp	9
2.1.2 Bài toán phân loại ảnh với số lượng lớp lớn	9
2.2 Những vấn đề thách thức	9
2.2.1 Dữ liệu lớn	9
2.2.2 Các phương pháp phân loại	10
2.2.3 Biểu diễn ảnh	10
2.2.4 Độ chính xác	10
2.2.5 Chi phí phân loại	11
2.2.6 Cân bằng giữa độ chính xác và chi phí thực hiện	11
2.3 Những công trình nghiên cứu liên quan	11
2.3.1 Hướng tiếp cận phẳng	11
2.3.2 Hướng tiếp cận dựa trên cấu trúc cây phân cấp	11
2.3.3 Hướng tiếp cận khác	12
2.4 Một số vấn đề thách thức còn tồn tại	12
2.5 Mục tiêu luận án	13

Chương 3.	PHÂN LOẠI DỰA TRÊN CẤU TRÚC CÂY PHÂN CẤP	14
3.1	Giới thiệu	14
3.2	Xây dựng cây phân cấp cân bằng	15
3.2.1	Tổng quan về cây phân cấp	15
3.2.2	Xây dựng cây phân cấp cân bằng dựa trên tất cả các ảnh và ảnh trung bình	15
3.2.3	Xây dựng cây phân cấp cân bằng dựa trên sự tương đồng giữa các lớp	20
3.2.4	So sánh tính hiệu quả của các phương pháp đề xuất	24
3.3	Duyệt cây dựa trên thông tin các nút	25
3.3.1	Các cách tiếp cận hiện có	25
3.3.2	Các mối quan hệ giữa các nút	25
3.3.3	Xây dựng mô hình chọn nhánh ứng viên	26
3.3.4	Quá trình thực hiện phân loại	27
3.3.5	Thí nghiệm	28
3.4	Tổng kết chương	28
Chương 4.	PHÂN LOẠI DỰA TRÊN BỘ PHÂN LỚP TIỀM ẨN	29
4.1	Giới thiệu	29
4.2	Ý tưởng chính của phương pháp đề xuất	29
4.3	Xây dựng các bộ phân lớp tiềm ẩn	30
4.3.1	Mô tả phương pháp	30
4.3.2	Thuật toán huấn luyện	32
4.3.3	Thuật toán phân loại	32
4.4	Thực nghiệm	34
4.4.1	Thiết lập thực nghiệm	34
4.4.2	Kết quả thực nghiệm	34
4.5	So sánh với các phương pháp phân loại dựa trên cây phân cấp đã đề xuất	37
4.6	Tổng kết chương	37
Chương 5.	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	38
5.1	Kết luận	38

5.1.1	Những kết quả đạt được	38
5.1.2	Những ưu điểm và hạn chế của các phương pháp đề xuất	38
5.2	Hướng phát triển	39

DANH MỤC CÔNG BỐ KHOA HỌC **40**

Chương 1 TỔNG QUAN

1.1 Giới thiệu tóm tắt về công trình nghiên cứu

Phân loại (còn được gọi là đa phân lớp) ảnh với số lượng lớp lớn (hàng trăm đến hàng chục nghìn lớp) là một trong những bài toán quan trọng trong lĩnh vực máy học và thị giác máy tính, bởi vì nó có nhiều ứng dụng trong thực tiễn như truy xuất ảnh hoặc video, tổ chức quản lý và phân lớp tự động nội dung ảnh hoặc video,... và là một thành phần chính trong các bài toán phức tạp hơn như trong bài toán hiểu và tự động gán nhãn, mô tả nội dung ảnh hoặc video bằng các khái niệm,...

Trong luận án này, nghiên cứu sinh nghiên cứu và phát triển một số thuật toán hiệu quả để giải quyết các vấn đề thách thức của bài toán phân loại ảnh với số lượng lớn các lớp. Cụ thể như sau:

1. Phát triển phương pháp **xây dựng cấu trúc cây phân cấp cân bằng có độ chính xác cao và hiệu quả về chi phí phân lớp.**

Cấu trúc cây cân bằng nhằm đảm bảo độ dài đường đi từ nút gốc đến các nút lá là tương đương nhau, giúp nâng cao tính hiệu quả về chi phí phân lớp. Để tăng độ chính xác phân lớp, tất cả các ảnh và ảnh trung bình của mỗi lớp được sử dụng khi xây dựng cây. Mục đích là khai thác yếu tố đa dạng của các ảnh trong mỗi lớp và tính đại diện cho mỗi lớp khi phân chia các lớp vào các nhánh con. Điều này giúp cải thiện khả năng dự đoán đúng của các bộ phân lớp tại mỗi nút của cây và làm tăng độ chính xác phân lớp. Cả hai yếu tố cân bằng và độ chính xác được xét kết hợp trong quá trình xây dựng cây. Kết quả thực nghiệm trên các tập dữ liệu chuẩn đã chứng minh phương pháp được đề xuất hiệu quả hơn so các phương pháp khác. Nội dung của các phương pháp này đã được công bố ở hội thảo ICIAP-2015 [CT.3] và trên tạp chí CVIU [CT.2]. Nội dung chi tiết của các đóng góp này được trình bày trong chương 3 của luận án.

2. Phát triển phương pháp **xây dựng cấu trúc cây phân cấp cân bằng dựa trên sự tương đồng giữa các lớp.** Để đảm bảo độ chính xác của các bộ phân lớp tại các nút, thì các lớp càng tương đồng với nhau nên được phân vào cùng một nút con. Sự tương đồng giữa hai lớp được xác định dựa trên phương pháp sum match kernel (được tính trực tiếp dựa trên các vector đặc

trung) mà không dựa vào các bộ phân lớp nhị phân OvA như các phương pháp khác. Mặc dù phương pháp này khá hiệu quả nhưng có độ phức tạp tính toán cao. Hạn chế này được khắc phục bằng cách sử dụng phương pháp ánh xạ đặc trưng để chuyển các đặc trưng qua không gian mới. Khi đó sự tương đồng giữa hai lớp được xác định chỉ dựa trên các ảnh trung bình của hai lớp trong không gian mới này.

Ngoài ra, để giảm chi phí thực hiện phân lớp, nghiên cứu sinh đã đề xuất phương pháp cân bằng số lượng lớp trong các nhánh con để tạo cấu trúc cây cân bằng. Nội dung nghiên cứu này đã được trình bày chi tiết trong chương 3 và được công bố trong ở hội thảo quốc tế ATC-2015 [CT.4] và tạp chí JCC-2016 [CT.1].

3. Phát triển phương pháp **phân loại dựa trên thông tin các nút** nhằm hạn chế vấn đề lan truyền lỗi và nâng cao độ chính xác phân lớp. Trong nghiên cứu này, nghiên cứu sinh đã đề xuất một cách tiếp cận mới bằng cách xét đến **mỗi quan hệ giữa nút ứng viên với từng nút con của nó và mỗi quan hệ giữa các nút ứng viên với nhau để lựa chọn nút ứng viên tiếp theo**. Các thực nghiệm được tiến hành trên cùng một cấu trúc cây đã chứng minh độ chính xác phân lớp theo phương pháp đề xuất đạt kết quả tốt hơn so với phương pháp duyệt khác. Nội dung chi tiết của phương pháp này được trình bày trong chương 3 và đã được công bố ở hội thảo ICIP-2016 [CT.6].
4. Đề xuất phương pháp **phân loại dựa trên các bộ phân lớp tiềm ẩn**. Một trong những giới hạn của cấu trúc cây phân cấp là sự điều chỉnh giữa độ chính xác và chi phí phân lớp sao cho có thể sử dụng tốt nhất tài nguyên có thể có của hệ thống để đạt được một độ chính xác tốt nhất. Lý do là vì cấu trúc cây đã được cố định sau khi xây dựng. Trong khi các phương pháp OvA có chi phí tuyến tính với số lớp. Trong luận án này, nghiên cứu sinh đã đề xuất một phương pháp mới dựa trên kĩ thuật xấp xỉ ma trận bậc thấp để **chuyển bài toán đa phân lớp trong không gian với số lượng lớp lớn về bài toán phân đa lớp trong không gian có số lượng lớp nhỏ hơn**. Từ các kết quả phân lớp trong không gian nhỏ hơn này, ta thực hiện phép kết hợp ma trận để tạo ra kết quả phân lớp trong không gian ban đầu. Bằng cách này, chúng

ta có thể chọn số bộ phân lớp tương ứng để sử dụng tốt nhất tài nguyên của hệ thống nhằm đạt được một độ chính xác tốt nhất. Nội dung nghiên cứu được trình bày chi tiết trong chương 4 và đã được công bố ở hội thảo MMSP-2015 [CT.5].

1.2 Động lực nghiên cứu

Bài toán phân loại ảnh với số lượng lớp lớn là một trong những bài toán quan trọng trong lĩnh vực thị giác máy tính do có nhiều ứng dụng như phát hiện đối tượng, phân lớp tự động, tổ chức quản lý và truy xuất ảnh-video, gán chú thích và mô tả nội dung ảnh-video,...

Tuy nhiên, độ phức tạp phân loại của các phương pháp thông thường (chẳng hạn như phương pháp OVA) tỉ lệ với số lượng lớp, vì thế trong trường hợp số lượng lớp lớn sẽ làm hạn chế khả năng ứng dụng của các phương pháp này trong thực tiễn.

Một số phương pháp phân loại dựa trên cấu trúc cây phân cấp có chi phí phân loại thấp, tuy nhiên do chỉ dùng một số lượng nhỏ các bộ phân lớp để phân loại ảnh, nên độ chính xác thường giảm so với phương pháp OVA. Ngoài ra, làm thế nào để xây dựng cây tối ưu và khắc phục việc lan truyền lỗi trong cây đang là những vấn đề thách thức.

1.3 Mục đích, đối tượng và phạm vi nghiên cứu

- **Mục đích nghiên cứu:** Nghiên cứu và phát triển các phương pháp phân lớp với số lượng lớp lớn hiệu quả về chi phí thực hiện phân lớp và đạt được độ chính xác phân lớp cao.
- **Đối tượng nghiên cứu:** Đối tượng nghiên cứu trong luận án là bài toán phân loại (đa phân lớp) trong trường hợp số lượng lớp lớn.
- **Phạm vi nghiên cứu:** Phạm vi của nghiên cứu này được giới hạn trên tập dữ liệu ảnh có số lượng lớp lớn. Trong đó:
 - Khái niệm lớp ảnh được sử dụng trong luận án được hiểu là một tập các ảnh có nội dung liên quan đến một khái niệm ngữ nghĩa được hiểu bởi con người. Các khái niệm thường được biểu diễn bằng các danh từ chung hoặc danh từ riêng với nhiều mức độ ngữ nghĩa và loại khác nhau (ví dụ như Object, Scene, Event, Artifacts, People, ...).

- Số lượng lớp (large-scale) được xét trong nghiên cứu này là từ vài trăm đến vài chục nghìn lớp. Các thực nghiệm được tiến hành trên các tập dữ liệu được cộng đồng nghiên cứu thừa nhận và sử dụng liên quan đến bài toán phân loại với số lượng lớp lớn.

1.4 Ý nghĩa khoa học và thực tiễn của đề tài

- Nội dung chính nghiên cứu sinh thực hiện trong luận án này là nghiên cứu và đề xuất một số phương pháp để giải quyết những thách thức trong bài toán phân loại ảnh với số lượng lớp lớn. Nghiên cứu sinh đã đề xuất phương pháp mới về bộ phân lớp tiềm ẩn và các phương pháp cải tiến trên tiếp cận cây phân cấp nhằm nâng cao độ chính xác và giảm chi phí tính toán khi thực hiện phân loại với số lượng lớp lớn. Thực nghiệm được tiến hành và kiểm chứng trên các tập dữ liệu chuẩn quốc tế, kết quả có tính xác thực cao. Các phương pháp đề xuất này đã được công bố trong các hội thảo và tạp chí chuyên ngành có uy tín và được phản biện độc lập.
- Hệ thống phân loại ảnh có độ chính xác cao và chi phí tính toán hợp lý sẽ có nhiều hỗ trợ hữu hiệu trong các ứng dụng thực tế như: truy vấn ảnh-video; tổ chức, quản lý dữ liệu ảnh-video; gán chú thích ảnh-video; hiểu và diễn đạt nội dung ảnh-video,...
- Hiện nay, các hệ phân lớp ảnh dựa vào mạng học sâu đạt được nhiều tiến bộ đáng kể và một số phương pháp như phương pháp One-versus-All (OvA) có thể được thực hiện song song. Tuy nhiên, chúng ta có thể thấy:
 - Mục tiêu chính của phương pháp OvA song song là giảm thời gian thực hiện phân lớp nhưng chi phí tính toán để thực hiện phân lớp cho một ảnh là không giảm. Với phương pháp được đề xuất trong luận án (chẳng hạn phương pháp dựa trên các bộ phân lớp tiềm ẩn), chúng ta không chỉ có thể giảm chi phí tính toán bằng cách điều chỉnh số lượng bộ phân lớp tiềm ẩn để đạt hiệu suất tối ưu mà còn có thể thực hiện song song như phương pháp OvA.

- Trong hệ phân lớp ảnh dựa vào mạng học sâu (như mạng AlexNet, VGG-VERYDEEP, GoogLeNet) thì chức năng của lớp mạng cuối sẽ thực hiện thao tác phân loại như phương pháp OvA, do đó chi phí phân loại cho một ảnh của lớp mạng này tỉ lệ tuyến tính với số lượng lớp ảnh. Ảnh hưởng lớn nhất của mạng học sâu hiện nay trong bài toán phân loại ảnh là về mặt học đặc trưng tự động, chứ không phải về phương pháp phân loại. Trong khi đó, mục tiêu chính của luận án là về phương pháp phân loại và chi phí tính toán phân loại. Các đặc trưng học sâu hoàn toàn có thể được tích hợp vào các phương pháp được đề xuất nhằm nâng cao hiệu quả phân loại.

1.5 Bố cục luận án

Nội dung luận án được bố cục gồm 5 chương và tài liệu tham khảo. Chương 1: Tổng quan, Chương 2: Cơ sở lý thuyết, Chương 3: Phân loại dựa trên cấu trúc cây phân cấp, Chương 4: Phân loại dựa trên bộ phân lớp tiềm ẩn và Chương 5: Kết luận và hướng phát triển.

Chương 2 CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu bài toán

2.1.1 Bài toán đa phân lớp

Bài toán đa phân lớp là bài toán xác định một mẫu x (chưa biết thuộc lớp nào) thuộc vào một trong C lớp được định nghĩa trước.

2.1.2 Bài toán phân loại ảnh với số lượng lớp lớn

Là một bài toán đa phân lớp được thực hiện trên tập dữ liệu ảnh có số lượng lớp lớn. Trong bài toán này:

- Một tập gồm C lớp ảnh được cho trước. Mỗi lớp ảnh sẽ chứa một tập các ảnh có nội dung liên quan đến một khái niệm ngữ nghĩa được hiểu bởi con người. Khái niệm lớp có thể được biểu diễn bằng các danh từ chung hoặc danh từ riêng với nhiều mức độ ngữ nghĩa khác nhau.
- Số lượng lớp C có giá trị từ vài trăm đến vài chục ngàn lớp.
- Làm thế nào để xác định một ảnh mới x (có chứa nội dung liên quan đến một trong C lớp) thuộc vào lớp nào trong C lớp.

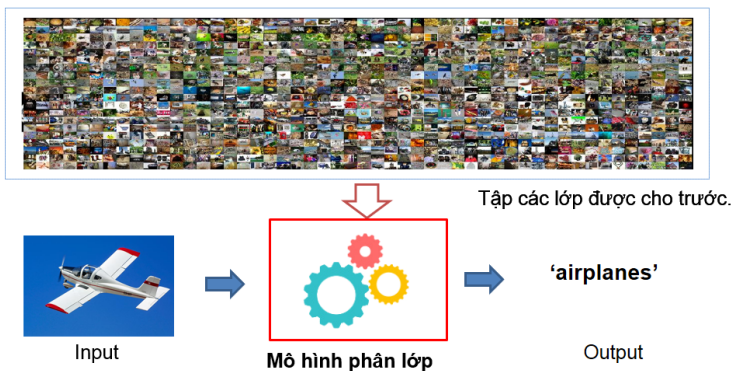
Hình 2.2 là một ví dụ minh họa cho bài toán phân loại ảnh trong trường hợp số lượng lớp lớn.

2.2 Những vấn đề thách thức

Có nhiều vấn đề thách thức liên quan đến bài toán phân loại với số lượng lớp lớn. Sau đây là một số thách thức quan trọng:

2.2.1 Dữ liệu lớn

- Khối lượng dữ liệu lớn đòi hỏi chúng ta phải có nhiều không gian lưu trữ, dung lượng bộ nhớ lớn và mất nhiều thời gian khi thực hiện như: thời gian để rút trích đặc trưng các ảnh, thời gian huấn luyện các bộ phân lớp, thời gian thực hiện phân loại,...
- Khi xét với số lượng lớp lớn, việc chia lớp chi tiết nên có nhiều lớp tương tự nhau, các ảnh trong cùng một lớp thường đa dạng hơn. Điều này làm cho bài toán phân lớp càng khó hơn.



Hình 2.2: Minh họa bài toán phân loại ảnh với số lượng lớp lớn. Trong trường hợp này, việc phân loại sẽ khó chính xác hơn, chi phí thực hiện phân loại sẽ cao hơn.

2.2.2 Các phương pháp phân loại

Một trong những phương pháp thường được sử dụng là phương pháp dựa trên các bộ phân lớp nhị phân OVA được hỗ trợ trong các thư viện như LIBSVM, LIBLINEAR,... Tuy nhiên các phương pháp này có độ phức tạp phân loại cao.

2.2.3 Biểu diễn ảnh

Có nhiều phương pháp biểu diễn ảnh được đề xuất và đạt kết quả tốt như mô hình BOW dựa trên đặc trưng cục bộ SIFT, phương pháp mã hóa LLC, phương pháp SPM.

Trong những năm gần đây, phương pháp học sâu (deep learning) đã đạt được những thành tựu đáng kể. Tuy nhiên, để có thể xây dựng một mô hình cho một bài toán cụ thể, quá trình thiết kế các lớp là một công việc phức tạp.

2.2.4 Độ chính xác

Khi số lượng các lớp càng lớn, tính phân biệt giữa các lớp trên không gian đặc trưng càng giảm đi, độ đa dạng của các ảnh trong các lớp càng cao,... thì việc xây dựng các mô hình có độ chính xác cao nhưng hiệu quả về chi phí thực hiện là một thách thức.

2.2.5 Chi phí phân loại

Chi phí phân loại với số lượng lớp lớn là một trong những thách thức chính khi áp dụng các phương pháp phân loại trong thực tiễn. Phương pháp OvA có chi phí tuyến tính với số lượng lớp. Phương pháp dựa trên cây phân cấp có chi phí logarithm với số lượng lớp nhưng độ chính xác không cao.

2.2.6 Cân bằng giữa độ chính xác và chi phí thực hiện

Làm thế nào để phương pháp phân loại có thể vừa đảm bảo tính chính xác cao nhưng vừa đảm bảo tính hiệu quả về chi phí thực hiện là một thách thức lớn hiện nay trong các phương pháp phân loại với số lượng lớn.

2.3 Những công trình nghiên cứu liên quan

Qua phân tích, đánh giá các công trình nghiên cứu liên quan, ta có thể chia thành các hướng tiếp cận như sau:

2.3.1 Hướng tiếp cận phẳng

- Trong hướng tiếp cận này mối quan hệ giữa các lớp không được xem xét.
- Phương pháp thông thường là chuyển bài toán phân loại nhiều lớp về các bài toán phân loại nhị phân và dựa trên các kết quả phân loại nhị phân để xác định kết quả phân loại nhiều lớp. Một số phương pháp phổ biến như: OvA, OvO, DAGSVM, ECOC,...
- Phương pháp OVA được sử dụng rộng rãi do tính đơn giản, số lượng bộ phân lớp bằng với số lớp và có nhiều thư viện hỗ trợ như LIBSVM và LIBLINEAR. Tuy nhiên, chi phí thực hiện phân loại phụ thuộc tuyến tính vào số lượng lớp. Trong trường hợp số lượng lớp lớn hoặc số lượng lớn ảnh cần phân loại thì các phương pháp này có tính khả thi thấp.

2.3.2 Hướng tiếp cận dựa trên cấu trúc cây phân cấp

- Trong hướng tiếp cận này có xét đến mối quan hệ giữa các lớp và tổ chức các lớp theo cấu trúc cây phân cấp nhằm mục tiêu giảm chi phí thực hiện phân loại.

- Các phương pháp phân loại dựa trên cấu trúc cây phân cấp thường sử dụng một số lượng bộ phân lớp ít hơn so với phương pháp OvA, vì thế nó hiệu quả về tính toán trong trường hợp số lượng lớp lớn hoặc số lượng lớn ảnh cần phân loại.
- Có hai vấn đề chính trong cách tiếp cận này: xây dựng cấu trúc cây và sử dụng cây để thực hiện phân lớp.
- Ngoài ra, một hạn chế của hướng tiếp cận này là vấn đề lan truyền lỗi (error propagation) khi duyệt cây trong quá trình phân loại: nếu kết quả phân loại tại nút ở mức trên bị sai thì kết quả phân loại tại các nút ở mức phía dưới hiển nhiên sẽ bị sai và không có cách khắc phục.

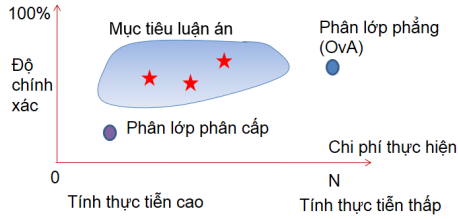
2.3.3 Hướng tiếp cận khác

Ngoài việc tiếp cận theo các phương pháp phẳng hay phân cấp. Một số phương pháp khác tập trung vào vấn đề phát triển các mô hình biểu diễn ảnh (mô hình BOW), các đặc trưng trong ảnh như đặc trưng cục bộ (SIFT), đặc trưng học sâu (CNN),... để nâng cao độ chính xác phân lớp.

2.4 Một số vấn đề thách thức còn tồn tại

Một số thách thức chính mà nội dung luận văn hướng đến giải quyết như sau:

- *Độ chính xác phân loại và chi phí thực hiện phân loại trong cấu trúc cây phân cấp.* Mục đích chính của các phương pháp sử dụng cấu trúc cây là để giảm chi phí thực hiện phân loại, tuy nhiên điều này làm cho độ chính xác phân loại cũng giảm theo. Việc phát triển các phương pháp hiệu quả hơn là rất cần thiết.
- *Tính đa dạng của các ảnh trong các lớp chưa được sử dụng trong quá trình phân chia các nhóm.* Quá trình xây dựng cấu trúc cây phân cấp là một quá trình phân chia một tập các lớp vào các nút con tương ứng. Việc khai thác tính đa dạng của các ảnh trong các lớp có thể giúp nâng cao độ chính xác phân nhóm các lớp.
- *Mối quan hệ tiềm ẩn giữa các lớp chưa được khai thác.* Khi số lớp ngày càng tăng, mối quan hệ về thị giác và ngữ nghĩa của các lớp càng lớn. Tuy nhiên, trong các nghiên cứu liên quan, các mối quan hệ này chưa được chú trọng trong quá trình phát triển các phương pháp.



Hình 2.9: Minh họa mục tiêu của luận án: phát triển các phương pháp phân loại ảnh hiệu quả về chi phí tính toán khi phân loại và đạt độ chính xác cao.

2.5 Mục tiêu luận án

Từ những công trình nghiên cứu liên quan và một số vấn đề thách thức còn tồn tại, chúng tôi đề ra mục tiêu chính của luận án là phát triển các phương pháp phân loại với số lượng lớp lớn sao cho các phương pháp này đạt hiệu quả về chi phí tính toán khi phân loại và có độ chính xác cao. Hình 2.9 minh họa cho mục tiêu của luận án.

Để đạt được mục tiêu này, luận án có hai hướng tiếp cận chính như sau:

- Hướng tiếp cận 1: phân loại dựa trên cấu trúc cây phân cấp. Đây là một trong những phương pháp phân loại hiệu quả về chi phí. Trong hướng cận này, có hai vấn đề chính:
 - Vấn đề xây dựng cấu trúc cây tối ưu (về chi phí thực hiện lẫn độ chính xác phân loại).
 - Vấn đề duyệt cây với độ lỗi thấp (giúp giảm vấn đề lan truyền lỗi và gia tăng độ chính xác phân loại).

Trong luận án, nghiên cứu sinh đã đề xuất một số phương pháp cải tiến để giải quyết hai vấn đề này. Nội dung được trình bày chi tiết trong chương 3.

- Hướng tiếp cận 2: phân loại dựa trên các lớp tiềm ẩn. Đây là một phương pháp mới được đề xuất trong luận án. Ý tưởng chính của hướng tiếp cận này là chỉ sử dụng một số lượng nhỏ các bộ phân lớp tiềm ẩn để thực hiện phân loại với số lượng lớp lớn.

Chương 3 PHÂN LOẠI DỰA TRÊN CẤU TRÚC CÂY PHÂN CẤP

Tóm tắt

Chương này trình bày về những đề xuất mới trong cách tiếp cận cây phân cấp (hierarchical label tree) dùng để giảm chi phí cho quá trình phân loại. Những đóng góp chính của nghiên cứu sinh gồm:

- Phát triển phương pháp xây dựng cây phân cấp cân bằng dựa trên tất cả các ảnh và ảnh trung bình. Các kết quả nghiên cứu được công bố trong kỉ yếu hội nghị quốc tế ICIAP 2015 (oral presentation, ERA-B) [CT.3] và tạp chí CVIU 2016 (ISI) [CT.2].
- Phát triển phương pháp xây dựng cây phân cấp cân bằng dựa trên sự tương đồng giữa các lớp. Nội dung của phương pháp đã được công bố trong kỉ yếu hội nghị quốc tế ATC 2015 [CT.4] và tạp chí Tin học và Điều khiển học - JCC [CT.1].
- Phát triển phương pháp duyệt cây dựa trên thông tin các nút để cải tiến độ chính xác phân loại. Các kết quả của các phương pháp này được công bố trong kỉ yếu hội nghị quốc tế ICIP 2016 (lecture(oral) presentation, ERA-B) [CT.6].

3.1 Giới thiệu

Có hai giai đoạn chính trong quá trình xây dựng cấu trúc cây:

- Giai đoạn 1: xây dựng cấu trúc cây. Trong giai đoạn này, các tiêu chí để phân nhóm các lớp và phương pháp phân nhóm được sử dụng để phân các lớp trong mỗi nút vào các nút con của nó.
- Giai đoạn 2: huấn luyện các bộ phân lớp tại các nút của cây dựa trên sự phân nhóm các lớp trong giai đoạn 1.

Trong nghiên cứu này, nghiên cứu sinh phát triển phương pháp xây dựng cấu trúc cây dựa trên hai yếu tố chính như sau:

- *Tính cân bằng*: để đạt được tính hiệu quả về chi phí tính toán khi thực hiện phân loại, đòi hỏi cấu trúc cây phải đảm bảo tính cân bằng, do đó trong quá trình phân các lớp vào các nút con phải xét đến sự cân bằng về số lượng lớp trong mỗi nút con.

- *Độ chính xác khi thực hiện phân nhóm các lớp*: trong quá trình xây dựng cấu trúc cây, các lớp dễ gây nhập nhằng với nhau hoặc càng giống nhau thì nên phân vào cùng một nút con. Điều này giúp các bộ phân lớp tại các nút dự đoán chính xác hơn, và giúp cải tiến độ chính xác phân loại.

3.2 Xây dựng cây phân cấp cân bằng

3.2.1 Tổng quan về cây phân cấp

Một cây phân cấp T_Q là một cấu trúc phân cấp của một tập L các lớp. Mỗi nút v trong cây chứa một tập các lớp $\ell(v) \subseteq L$ và có tối đa Q nút con $\sigma(v) = \{\zeta_1, \dots, \zeta_Q\}$. Nút gốc chứa tất cả các lớp $\ell(v = \text{root}) = L$ và mỗi nút lá chứa một lớp $\ell(v = \text{leaf}) \subseteq L, |\ell(v = \text{leaf})| = 1$.

Quá trình xây dựng cấu trúc cây thường được thực hiện đệ quy bằng cách phân một tập các lớp vào các nhóm, mỗi nhóm tương ứng với một nút con, bắt đầu từ nút gốc, cho đến khi cấu trúc cây được tạo thành.

Sau khi có được một cây phân cấp T_Q , ta có phân loại cho một ảnh x bằng cách duyệt cây từ nút gốc cho đến khi đạt đến nút lá. Ảnh x sẽ được phân vào lớp tương ứng của nút lá này.

3.2.2 Xây dựng cây phân cấp cân bằng dựa trên tất cả các ảnh và ảnh trung bình

3.2.2.1 Điều kiện xây dựng cấu trúc cây cân bằng

Để tạo cấu trúc cây phân cấp cân bằng $T_{Q,H}$ sao cho mỗi nút của cây có tối đa Q nhánh và chiều cao tối đa là H , thì chúng ta cần phải xét đến số lượng các lớp được phân vào các nút con. Giả sử, nút v có $|\ell(v)|$ lớp thì mỗi nút con của v sẽ có tối đa $P(v)_{max}$ lớp:

$$P(v)_{max} = Q^{h(v)-1} \quad (3.1)$$

trong đó giá trị $h(v) = \log_Q(|\ell(v)|)$ là độ cao tối đa có thể có tương ứng với số lượng $|\ell(v)|$ lớp.

Gọi ma trận $S(v)_{|\ell(v)| \times |\sigma(v)|}$ chứa thông tin về sự phân $|\ell(v)|$ lớp

vào các nút con. Giá trị của $S(v)_{i,j}$ có ý nghĩa như sau:

$$S(v)_{i,j} = \begin{cases} 1, & \text{nếu lớp thứ } i \text{ thuộc vào nút con thứ } j : \\ & c_i \in \ell(v) \text{ và } c_i \in \ell(\zeta_j), \zeta_j \in \sigma(v) \\ 0, & \text{ngược lại} \end{cases} \quad (3.2)$$

Giả sử mỗi lớp chỉ thuộc vào một nhóm. Khi đó ta có thể mô tả điều kiện này cho lớp thứ i của v như sau:

$$\sum_{j=1}^{|\sigma(v)|} S(v)_{i,j} = 1, \quad (3.3)$$

Điều kiện nhóm thứ j chứa tối đa $P(v)_{max}$ lớp được mô tả như sau:

$$\sum_{i=1}^{|\ell(v)|} S(v)_{i,j} \leq P(v)_{max} \quad (3.4)$$

3.2.2.2 Phân nhóm các lớp dựa trên tất cả các ảnh

Việc phân các lớp vào các nút con được thực hiện dựa trên trung bình khoảng cách từ tất cả các ảnh trong một lớp đến phần tử tâm của các nhóm. Điều này sẽ giúp tăng độ chính xác của quá trình phân các lớp. Chúng ta gọi:

- \tilde{C}_j là phần tử tâm của nhóm thứ j , mỗi nhóm tương ứng với một nút con.
- $d(x, \tilde{C}_j)$ là một hàm đo khoảng cách từ vector đặc trưng x đến tâm \tilde{C}_j của nhóm thứ j .
- $x_{i,k}$ là vector đặc trưng của ảnh thứ k thuộc lớp thứ i .
- $F(v)_{|\ell(v)| \times |\sigma(v)|}$ là ma trận chứa thông tin về khoảng cách từ $|\ell(v)|$ lớp đến tâm của $|\sigma(v)|$ nhóm khi ta xét tại nút v .

Giá trị $F(v)_{i,j}$ được tính bằng trung bình khoảng cách từ tất cả các vector đặc trưng của các ảnh thuộc lớp i đến tâm của nhóm thứ j như

sau:

$$F(v)_{i,j} = \frac{1}{n_i} \sum_{k=1}^{n_i} d(x_{i,k}, \tilde{C}_j) \quad (3.5)$$

Nếu lớp thứ i thuộc vào nhóm thứ j thì giá trị của $F(v)_{i,j}$ là giá trị nhỏ nhất trong tất cả các giá trị $\{F(v)_{i,1}, \dots, F(v)_{i,|\sigma(v)|}\}$. Điều này cũng có nghĩa là các lớp thuộc cùng một nhóm thứ j là các lớp có khoảng cách $F(v)_{i,j}$ nhỏ nhất. Nói cách khác thì tổng khoảng cách của các lớp thuộc vào nhóm thứ j là nhỏ nhất:

$$\min_{\ell(j)} \sum_{i=1}^{|\ell(j)|} F(v)_{i,j} \quad (3.6)$$

Cách tiếp cận này đã được công bố trong kỉ yếu hội nghị quốc tế ICIAP 2015 [CT.3].

3.2.2.3 Phân nhóm các lớp dựa trên tất cả các ảnh và ảnh trung bình

Việc sử dụng tất cả các phần tử của các lớp để thực hiện phân nhóm sẽ tận dụng được các yếu tố đặc trưng của các lớp đó, tuy nhiên, cách tiếp cận này có hạn chế là nhạy cảm với các phần tử ở biên của lớp. Trong khi cách tiếp cận dựa trên phần tử trung bình có khả năng xử lý được các phần tử biên nhưng không đảm bảo được tính đại diện trong các lớp có mức độ đa dạng lớn. Chúng tôi kết hợp ưu điểm của cách tiếp cận này để thực hiện phân nhóm các lớp khi xây dựng cấu trúc cây. Khi đó công thức (3.5) xác định $F(v)_{i,j}$ được mở rộng như sau:

$$F(v)_{i,j} = \frac{1}{n_i} \sum_{k=1}^{n_i} d(x_{i,k}, \tilde{C}_j) + d(\tilde{x}_i, \tilde{C}_j), \quad (3.7)$$

trong đó $\tilde{x}_i = \frac{1}{n_i} \sum_{k=1}^{n_i} x_{i,k}$ là vector đặc trưng trung bình của n_i ảnh của lớp i . Công thức (3.7) là một sự kết hợp giữa việc sử dụng ảnh trung bình (thích hợp với các lớp mà ảnh trong lớp đó phân bố quanh tâm của lớp) và sử dụng tất cả ảnh của lớp (thích hợp với các lớp có đa số ảnh phân tán).

Cách tiếp cận này đã được công bố trong tạp chí CVIU [CT.2].

3.2.2.4 Xây dựng cấu trúc cây cân bằng

Để xây dựng một cấu trúc cây cân bằng và các bộ phân lớp tại mỗi nút có độ chính xác cao, tại mỗi nút v ta cần đảm bảo các điều kiện về số lượng các lớp trong mỗi nút con và khoảng cách giữa các lớp trong cùng một nhóm đến tâm của nhóm đó phải nhỏ nhất. Đây là bài toán tối ưu: tìm các giá trị của ma trận $S(v)_{|\ell(v)| \times |\sigma(v)|}$ và ma trận $F(v)_{|\ell(v)| \times |\sigma(v)|}$ sao cho với các giá trị trong $S(v)$ thì ma trận khoảng cách $F(v)$ của các lớp trong cùng một nhóm là nhỏ nhất. Do đó, ta có bài toán như sau:

$$\min_{S(v), F(v)} \sum_{i=1}^{|\ell(v)|} \sum_{j=1}^{|\sigma(v)|} S(v)_{i,j} \cdot F(v)_{i,j}, \quad (3.8)$$

với điều kiện cân bằng (3.4) và các điều kiện về giá trị của $S(v)$ là (3.2) và (3.3). Trong đó giá trị của $F(v)_{i,j}$ được xác định theo công thức (3.5) hoặc (3.7).

Bài toán (3.8) là một bài toán tối ưu bi-linear với hai biến không âm là $S(v)$ và $F(v)$. Bài toán này có thể được giải bằng phương pháp tối ưu thay thế trong hai bước (two alternating convex optimizations): cố định giá trị $F(v)$ để tìm giá trị $S(v)$, sau đó cố định giá trị $S(v)$ để tìm giá trị $F(v)$.

Để xây dựng một cấu trúc cây cân bằng $T_{Q,H}$, ta bắt đầu từ nút gốc của cây, áp dụng thuật toán 3.1 để thực hiện phân nhóm các lớp tại mỗi nút của cây. Quá trình này được thực hiện một cách đệ quy cho đến khi nào cấu trúc cây được hoàn thiện.

Trong thực nghiệm, ký hiệu BLTree-A tương ứng với cây phân cấp cân bằng được xây dựng bằng cách sử dụng tất cả các ảnh (công thức 3.5) và ký hiệu BLTree-AM tương ứng với cây phân cấp cân bằng được xây dựng bằng cách kết hợp tất cả các ảnh và ảnh trung bình (công thức 3.7).

3.2.2.5 Thí nghiệm

Thực nghiệm được tiến hành trên các tập dữ liệu chuẩn Caltech-256, SUN-397, ILSVRC2010-1K và Imagenet-10K. Hai đặc trưng

Algorithm 3.1 $[A] = \text{SplittingBalancing}(\ell(v), X, Q, P(v)_{max}, t)$:
phân tập các lớp $\ell(v)$ vào Q nhóm và thực hiện cân bằng số lượng lớp
trong mỗi nhóm. Mỗi nhóm tương ứng với một nút con của v .

Input:

- 1: $\ell(v)$: tập các lớp của nút v ;
- 2: $X = \{(x_i, y_i)\}$: tập ảnh của các lớp tại nút v với $\cup y_i = \ell(v)$;
- 3: Q : số nút con (số nhóm) tối đa của nút v ;
- 4: $P(v)_{max}$: số lượng lớp tối đa trong mỗi nhóm;
- 5: t : Số lần lặp tối đa khi tìm lời giải tối ưu;

Output:

- 6: $A = \{a_1, \dots, a_N\}$: là tập hợp gồm N phần tử, mỗi phần tử $a_i = k$ sẽ cho biết thông tin lớp $c_i \in \ell(v)$ được phân vào nhóm thứ k ; Số lượng lớp tối đa trong mỗi nhóm là $P(v)_{max}$.
 - 7: **Bước 1:** Khởi tạo các vector tâm C_Q của Q nhóm: $C_Q = k\text{-means}(X, Q)$.
 - 8: **Bước 2:** Tính ma trận $F(v)$ sử dụng phương trình (3.5) hoặc (3.7).
 - 9: **Bước 3:** Tìm ma trận $S(v)$: cố định giá trị $F(v)$, giải phương trình (3.8) để tìm $S(v)$ theo các điều kiện (3.2), (3.3) và (3.4).
 - 10: **Bước 4:** Cập nhật lại các giá trị C_Q dựa trên thông tin tìm được trong ma trận $S(v)$.
 - 11: **Bước 5:** Lặp lại Bước 2 đến khi lời giải của (3.8) hội tụ hoặc đã đạt được t lần lặp.
-

được sử dụng là BOW-SIFT-LLC-SPM có kích thước 50.000 chiều (đây là đặc trưng được sử dụng phổ biến trong các công trình nghiên cứu liên quan trước đây) và VGG-VERYDEEP-16 có kích thước 4.096 chiều (đây là một đặc trưng học sâu và cho kết quả cao trong nhiều bài toán xử lý ảnh).

Quan sát kết quả trên các cấu hình cây khác nhau, chúng ta có nhận xét tổng quát sau:

- Khi cây càng cao, độ dài đường đi từ nút gốc đến nút lá sẽ dài ra, số phép toán dot-products của các bộ phân lớp tại các nút sẽ giảm, độ tăng tốc S_{te} càng tăng, tuy nhiên, độ chính xác Acc lại giảm vì có ít bộ phân lớp được dùng để phân loại.
- Kết quả thực nghiệm trên các tập dữ liệu chuẩn đã chứng minh tính hiệu quả của phương pháp đề xuất so với các phương pháp liên quan. Tại cùng giá trị chính xác Acc thì phương pháp đề xuất hiệu quả hơn (S_{te} lớn hơn), và tại cùng một giá trị S_{te} thì độ chính xác của phương pháp đề xuất cao hơn.
- Ngoài ra, với cùng số lượng bộ phân lớp được sử dụng như trong cách tiếp cận dùng cấu trúc cây, độ chính xác của các phương pháp trong cách tiếp cận ECOOC đều thấp hơn.

Qua thực nghiệm, chúng ta có thể thấy việc sử dụng tất cả các ảnh và ảnh trung bình của từng lớp có thể giúp cải tiến độ chính xác.

3.2.3 Xây dựng cây phân cấp cân bằng dựa trên sự tương đồng giữa các lớp

3.2.3.1 Ma trận tương đồng

Một cách tiếp cận khác để phân các lớp vào các cây con là dựa trên ma trận tương đồng giữa các lớp.

Gọi \tilde{S} là ma trận thể hiện mức độ tương đồng giữa các lớp. Mỗi phần tử $\tilde{S}_{i,j}$ thể hiện mức độ tương đồng giữa lớp thứ i và lớp thứ j .

Giá trị $\tilde{S}_{i,j}$ được xác định bằng phương pháp sum-match kernel như sau:

$$\tilde{S}_{i,j} = \frac{1}{n_i} \frac{1}{n_j} \sum_{p=1}^{n_i} \sum_{q=1}^{n_j} k(f_{i,p}, f_{j,q}) \quad (3.9)$$

trong đó $k(\cdot)$ là một hàm nhân Mercer; n_i và n_j tương ứng là tổng số

ảnh có trong các lớp thứ i và thứ j ; $f_{i,p}, f_{j,q}$ là các vector đặc trưng của các ảnh trong lớp thứ i và thứ j .

Kết quả sử dụng cách tính này đã được công bố ở hội nghị quốc tế ATC 2015 [CT.4]. Tuy nhiên, điểm hạn chế là độ phức tạp tính toán lớn, ta phải thực hiện $n_i \times n_j$ lần tính hàm nhân $k(\cdot)$ giữa các cặp ảnh thuộc hai lớp thứ i và thứ j . Để khắc phục hạn chế này, nghiên cứu sinh đã đề xuất một hướng tiếp cận dựa trên phương pháp ánh xạ đặc trưng và được trình bày trong phần 3.2.3.2.

3.2.3.2 Chuyển đổi không gian đặc trưng

Theo tính chất tái tạo (reproducing kernel) trong không gian Hilbert, luôn luôn tồn tại một ánh xạ φ vào không gian Hilbert \mathcal{H} cho bất kỳ hàm nhân định nghĩa dương $k(x, y)$ như sau:

$$k(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}} \quad (3.10)$$

trong đó $\varphi(x)$ và $\varphi(y)$ là các điểm dữ liệu trong không gian Hilbert \mathcal{H} , nó được ánh xạ từ hai điểm x và y , và $\langle \varphi(x), \varphi(y) \rangle$ kí hiệu cho phép tính tích giữa hai vector $\varphi(x)$ và $\varphi(y)$.

3.2.3.3 Tính sự tương đồng giữa hai lớp

Sau khi áp dụng phương pháp chuyển đổi không gian đặc trưng, hàm nhân $k(x, y)$ có thể được tính bằng hàm tuyến tính trong không gian Hilbert. Ta có:

$$k(x, y) = \langle \varphi(x), \varphi(y) \rangle = \varphi(x)^T \cdot \varphi(y) \quad (3.11)$$

Khi đó giá trị $\tilde{S}_{i,j}$ trong công thức (3.9) có thể được tính như sau:

$$\begin{aligned} \tilde{S}_{i,j} &= \frac{1}{n_i} \frac{1}{n_j} \sum_{p=1}^{n_i} \sum_{q=1}^{n_j} k(f_{i,p}, f_{j,q}) = \frac{1}{n_i} \frac{1}{n_j} \sum_{p=1}^{n_i} \sum_{q=1}^{n_j} (\varphi(f_{i,p})^T \cdot \varphi(f_{j,q})) \\ &= \frac{1}{n_i} (\varphi(f_{i,1}) + \dots + \varphi(f_{i,n_i}))^T \cdot \frac{1}{n_j} (\varphi(f_{j,1}) + \dots + \varphi(f_{j,n_j})) \\ &= \tilde{\varphi}_i^T \cdot \tilde{\varphi}_j \end{aligned} \quad (3.12)$$

trong đó $\tilde{\varphi}_i = \frac{1}{n_i}(\varphi(f_{i,1}) + \dots + \varphi(f_{i,n_i}))$ và $\tilde{\varphi}_j = \frac{1}{n_j}(\varphi(f_{j,1}) + \dots + \varphi(f_{j,n_j}))$ là các vector đặc trưng trung bình trong không gian đã chuyển đổi của hai lớp thứ i và thứ j .

Chi phí để tính $\tilde{S}_{i,j}$ dựa trên công thức (3.12) chỉ là phép tính tích giữa hai vector trong không gian mới, chi phí này thấp hơn so với cách tính theo công thức (3.9).

3.2.3.4 Quá trình phân nhóm các lớp để tạo cây cân bằng

Để tạo một cấu trúc cây cân bằng, tại mỗi nút v , các nút con của nó có ít nhất $P(v)_{max} = Q^{H-1}$ lớp, trong đó $H = \log_Q(N)$ là độ sâu tối đa của nút v tính từ nút gốc, N là số lượng lớp thuộc nút v . Vì vậy, nếu số lượng lớp trong nút con thứ j lớn hơn $P(v)_{max}$, thì nó cần phải được điều chỉnh. Quá trình điều chỉnh được trình bày tóm tắt trong thuật toán 3.2.

3.2.3.5 Xây dựng cấu trúc cây cân bằng

Có hai giai đoạn chính: đầu tiên là tính ma trận tương đồng $\tilde{S}_{N \times N}$ cho từng lớp theo công thức (3.12) đã được trình bày trong phần 3.2.3.3 (ma trận này chỉ tính một lần); tiếp theo là dựa vào ma trận tương đồng giữa các lớp tại mỗi nút của cây (bắt đầu từ nút gốc chứa tất cả N lớp) để thực hiện phân các lớp vào các nhóm và áp dụng thuật toán 3.2 để điều chỉnh các lớp trong các nhóm để cân bằng số lượng các lớp trong mỗi nút con nhằm mục đích xây dựng một cấu trúc cây cân bằng. Toàn bộ quá trình này được trình bày tóm tắt trong thuật toán 3.3.

Để xây dựng một cấu trúc cây, ta áp dụng thuật toán này một cách lặp lại cho từng nút của cây, bắt đầu từ nút gốc, cho đến khi cấu trúc cây được xây dựng hoàn chỉnh.

3.2.3.6 Thí nghiệm

Thực nghiệm được tiến hành trên các tập dữ liệu chuẩn Caltech-256, SUN-397, ILSVRC2010-1K và Imagenet-10K. Hai đặc trưng được sử dụng là BOW-SIFT-LLC-SPM và VGG-VERYDEEP-16. Quá trình thực nghiệm sử dụng một số hàm nhân được ứng dụng phổ biến trong lĩnh vực thị giác máy tính như χ^2 (BLTree-SMK-*kchi2*), *Intersection* (BLTree-SMK-*kinters*) và *Jensen – Shannon* (BLTree-SMK-*kjs*).

Từ các kết quả thực nghiệm, chúng ta có thể rút ra một số kết luận quan trọng như sau:

Algorithm 3.2 $[A] = \text{Balancing}(\mathcal{Y}, G, A, P(v)_{max})$: cân bằng số lượng lớp trong mỗi nút con của nút v

Input:

- 1: • $\mathcal{Y} = \{y_1, \dots, y_N\}$: là tập hợp gồm N vector đặc trưng biểu diễn cho N lớp tương ứng tại nút v ;
- 2: • $G = \{g_1, \dots, g_Q\}$: là tập hợp gồm Q giá trị tâm của Q nhóm;
- 3: • $A = \{a_1, \dots, a_N\}$: là tập hợp gồm N phần tử, mỗi phần tử $a_i = k$ sẽ cho biết thông tin lớp c_i được phân vào nhóm g_k ;
- 4: • $P(v)_{max}$: Số lớp tối đa trong mỗi nhóm;

Output: $A = \{a_1, \dots, a_N\}$: là tập hợp gồm N phần tử, mỗi phần tử $a_i = k$ sẽ cho biết thông tin lớp c_i được phân vào nhóm g_k ; Số lượng lớp tối đa trong mỗi nhóm là $P(v)_{max}$.

5: **Bước 1:**

- 6: • Gọi R là tập các nhóm có số lượng lớp lớn hơn $P(v)_{max}$.
- 7: • Gọi T là tập các nhóm có số lượng lớp nhỏ hơn $P(v)_{max}$.
- 8: • Gọi D là tập các phần tử sẽ được chọn để phân vào các nhóm trong T : $D = \emptyset$

9: **Bước 2:** Xét từng nhóm trong R : chỉ giữ lại $P(v)_{max}$ phần tử có khoảng cách đến tâm của nhóm đó là nhỏ nhất. Các phần tử còn lại sẽ thêm vào D .

10: **Bước 3:**

11: **while** $D \neq \emptyset$ **do**

12: $y_i \leftarrow D$

13: Phần tử y_i sẽ được phân vào nhóm $t_j \in T$ nếu khoảng cách từ y_i đến tâm g_j của nhóm t_j là nhỏ nhất: $t_j = t_j \cup \{y_i\}$

14: Cập nhật lại giá trị tâm g_j dựa trên các phần tử trong nhóm t_j .

15: **if** $|t_j| = P(v)_{max}$ **then**

16: $T = T \setminus \{t_j\}$

17: **end if**

18: **end while**

Algorithm 3.3 $[A] = Clustering(\ell(v), \tilde{S}_{N \times N}, Q, P(v)_{max})$: phân nhóm một tập các lớp $\ell(v)$ vào Q nút con.

Input:

- 1: • $\ell(v)$: tập hợp các lớp thuộc nút v ;
- 2: • $\tilde{S}_{N \times N}$: ma trận tương đồng giữa N lớp;
- 3: • Q : số nút con tối đa;
- 4: • $P(v)_{max}$: số lượng lớp tối đa trong mỗi nút con;

Output: $A = \{a_1, \dots, a_N\}$: là tập hợp gồm N phần tử, mỗi phần tử $a_i = k$ sẽ cho biết thông tin lớp c_i được phân vào nhóm g_k ; Số lượng lớp tối đa trong mỗi nhóm là $P(v)_{max}$.

- 5: **Bước 1:** Xác định ma trận tương đồng $\tilde{S}(v)$ dựa vào ma trận \tilde{S} và $\ell(v)$.
 - 6: **Bước 2:** Thực hiện phân nhóm các lớp:
 - 7: $[\mathcal{Y}, G, A] = SpectralClustering(\ell(v), \tilde{S}(v), Q, P(v)_{max})$
 - 8: **Bước 3:** Cân bằng số lượng lớp trong mỗi nút con: $[A] = Balancing(\mathcal{Y}, G, A, P(v)_{max})$
-

- Thứ nhất, độ chính xác phân loại (Acc) và độ tăng tốc (S_{te}) phụ thuộc vào cấu hình cây.
- Thứ hai, hiệu quả của phương pháp đề xuất tốt hơn so với các phương pháp liên quan khác trong hầu hết các trường hợp.
- Thứ ba, thời gian thực hiện phân loại của phương pháp đề xuất nhanh hơn so với những phương pháp khác trong hầu hết các cấu hình cây.

3.2.4 So sánh tính hiệu quả của các phương pháp đề xuất

Để so sánh hiệu quả của các phương pháp được đề xuất trong phần 3.2.2 và phần 3.2.3, chúng tôi đã tiến hành thực nghiệm hai phương pháp trên cùng tập dữ liệu ILSVRC2010-1K, gồm 1000 lớp, sử dụng đặc trưng VGG-VERYDEEP-16. Các kết quả được liệt kê trong bảng 3.9.

Dựa trên bảng kết quả so sánh, chúng ta có thể thấy cấu trúc cây được xây dựng theo phương pháp dựa trên tất cả các ảnh và ảnh trung bình có tính hiệu quả cao hơn các phương pháp khác. Trong phương pháp sử dụng ma trận tương đồng, chỉ có ảnh trung bình trong không gian mới được sử dụng, vì thế nó có ít thông tin để đưa ra lựa chọn phân nhóm đúng so với phương pháp dùng tất cả các ảnh và ảnh trung

Bảng 3.9: So sánh hiệu quả của các phương pháp khi dùng toàn bộ các ảnh huấn luyện được cung cấp trong tập dữ liệu ILSVRC2010-1K sử dụng đặc trưng VGG-VERYDEEP-16.

Phương pháp	$T_{32,2}$		$T_{10,3}$		$T_{6,4}$		$T_{4,5}$	
	Acc	S_{te}	Acc	S_{te}	Acc	S_{te}	Acc	S_{te}
Bengio et al.	53.21	16.01	44.27	32.59	40.06	39.64	36.00	45.42
Liu et al.	56.91	15.84	54.27	28.21	52.71	37.90	51.66	42.60
BLTree-SMK- <i>kchi2</i>	57.77	15.77	54.40	33.33	51.97	43.10	50.39	50.15
BLTree-SMK- <i>kinters</i>	57.82	15.74	54.26	33.33	52.75	43.42	50.09	50.15
BLTree-SMK- <i>kjs</i>	57.87	15.74	54.38	33.33	52.70	43.30	50.55	50.13
BLTree-A	58.02	15.73	54.55	33.33	52.05	42.65	51.08	50.11
BLTree-AM	58.13	15.76	54.84	33.33	52.91	42.46	51.54	50.09
OvA	55.73	1.0						

bình.

3.3 Duyệt cây dựa trên thông tin các nút

3.3.1 Các cách tiếp cận hiện có

Phương pháp cơ bản khi duyệt cây là nút được chọn tiếp theo là nút có giá trị dự đoán cao nhất. Tuy nhiên, nếu một nút v là nút được chọn sai, thì bất kỳ quyết định nào khi chọn nút con của v để duyệt cũng đều sai và không thể khắc phục.

Trong nghiên cứu này, chúng tôi khai thác các thông tin về mối quan hệ giữa các nút ứng viên để đưa ra quyết định chọn nút tiếp theo. Trong đó, giá trị thể hiện mối quan hệ được thể hiện qua giá trị dự đoán của các bộ phân lớp tại các nút tương ứng.

3.3.2 Các mối quan hệ giữa các nút

Chúng ta lần lượt gọi: $p_v(x)$ là giá trị dự đoán của bộ phân lớp p_v của nút v khi áp dụng trên ảnh x ; $\ell(v)$ là một tập các lớp thuộc vào nút v ; $\psi(v)$ là tập các nút con của v ; $v_i \in \psi(v)$ là nút con thứ i của v ; $v_{i,j} \in \psi(v_i)$ là nút con thứ j của nút con v_i .

Giả sử chúng ta đang xét nút v , nhiệm vụ là làm thế nào để chọn nút con $v_i \in \psi(v)$ để duyệt tiếp theo. Gọi $\rho_{v_i}(x) \in \mathbb{R}^{|\psi(v)|}$ là một vector được tạo thành bằng cách kết hợp giá trị dự đoán $p_{v_i}(x)$ và sự sai khác

giữa $p_{v_i}(x)$ với các nút con còn lại $v_k \in \psi(v), k \neq i$. Vector này có dạng như sau:

$$\rho_{v_i}(x) = [p_{v_i}(x), p_{v_i}(x) - p_{v_1}(x), \dots, p_{v_i}(x) - p_{v_{|\psi(v)|}}(x)] \quad (3.13)$$

Khi đó, một vector $\varphi_{i,j}(x)$ mô tả mối quan hệ giữa nút con v_i với các nút anh em và giữa v_i và các nút con của v_i được biểu diễn như sau:

$$\varphi_{i,j}(x) = [\rho_{v_i}(x), \rho_{v_{i,j}}(x)] \quad (3.14)$$

Vector $\varphi_{i,j}(x)$ cũng là một sự biểu diễn cho nhánh ứng viên từ nút v_i đến nút $v_{i,j}$.

3.3.3 Xây dựng mô hình chọn nhánh ứng viên

Với mỗi nút v_i là nút con của nút v , chúng ta có một tập các nhánh ứng viên đi qua nút v_i :

$$\gamma_{v_i}(x) = \{\varphi_{i,1}(x), \varphi_{i,2}(x), \dots, \varphi_{i,|\psi(v_i)|}(x)\} \quad (3.15)$$

Theo cách này, tại nút v ta sẽ có một danh sách các tập nhánh ứng viên $\gamma_{v_i}(x)$. Bài toán chọn nút con tiếp theo v_i có thể được giải dưới dạng tối ưu max margin.

Cho trước ảnh x đã được gán nhãn y . Một nhánh ứng viên được biểu diễn bằng một vector $\varphi_{i,j}(x)$. Vector này sẽ được gán nhãn dương nếu $y \in \ell(v_i)$ và $y \in \ell(v_{i,j})$, ngược lại, vector sẽ được gán nhãn âm.

Một tập $\gamma_{v_i}(x)$ chứa tất cả các nhánh ứng viên đi qua v_i được gán nhãn dương nếu nó có chứa ít nhất một nhánh ứng viên dương. Ngược lại, tập $\gamma_{v_i}(x)$ được gán nhãn âm.

Gọi Γ_v^+ là danh sách chứa tất cả các tập $\gamma_{v_i}(\cdot)$ có nhãn dương và Γ_v^- là danh sách chứa tất cả các tập $\gamma_{v_i}(\cdot)$ có nhãn âm. Khi đó, chúng ta có thể tìm hàm margin với các tham số (w_v, b_v) để tối đa hóa khoảng cách giữa Γ_v^+ và Γ_v^- .

Hàm mục tiêu sẽ được biểu diễn dưới dạng công thức tổng quát sử

dạng soft margin SVM như sau:

$$\begin{aligned} & \text{minimize } \frac{1}{2} \|w_v\|^2 + C \sum_i \xi_i^k \\ & \text{subject to } y_i(w_v^T x_i + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \end{aligned}$$

trong đó C là một giá trị hằng, $\{\xi_i\}$ là các biến không âm.

Để hạn chế mức độ ảnh hưởng của các outlier/error có hệ số ξ lớn trong tập huấn luyện, ta chọn giá trị $k = 1$, đây là trường hợp 1-norm soft margin. Khi đó công thức trên được viết lại như sau:

$$\min_{w_v, b_v, \{\xi_i\}} \frac{1}{2} \|w_v\|^2 + C \sum_i \xi_i \quad (3.16)$$

theo các điều kiện:

$$\max_{\gamma \in \Gamma_v^+} (w_v^T \cdot \gamma + b_v) \geq 1 - \xi_i, \quad \forall i = 1, \dots, |\Gamma_v^+|, \xi_i \geq 0 \quad (3.17)$$

$$\max_{\gamma \in \Gamma_v^-} (w_v^T \cdot \gamma + b_v) \leq -1 + \xi_j, \quad \forall j = 1, \dots, |\Gamma_v^-|, \xi_j \geq 0 \quad (3.18)$$

Bài toán tối ưu này có thể được giải theo phương pháp multiple-instance learning (MIL). Kết quả ta có được một mô hình tương ứng với hai tham số (w_v, b_v) . Mô hình này sẽ được sử dụng để ước lượng nút ứng viên tương ứng.

3.3.4 Quá trình thực hiện phân loại

Phân loại ảnh x bằng cách duyệt cây bắt đầu từ nút gốc đến nút lá. Tại mỗi nút v trên đường đi, ta tính được một danh sách Γ_v^x chứa $|\psi(v)|$ tập các nhánh ứng viên đi qua các nút $v_i, v_i \in \psi(v)$.

$$\Gamma_v^x = \{\gamma_{v_1}(x), \dots, \gamma_{v_{|\psi(v)|}}(x)\} \quad (3.19)$$

Sử dụng mô hình tương ứng với lời giải của phương trình (3.16) tại nút v , ta sẽ chọn được nút tiếp theo là nút có giá trị dự đoán lớn nhất:

$$\tilde{\gamma} = \arg \max_{\gamma \in \Gamma_v^x} (w_v^T \cdot \gamma + b_v) \quad (3.20)$$

Bảng 3.10: So sánh hiệu quả của các phương pháp trên tập dữ liệu ILSVRC2010-1K.

Phương pháp	T_{32}	T_{10}	T_6	T_4
Baseline	7.32	6.01	5.52	5.12
ER-SHC	7.70	5.70	5.12	4.66
Traverse-MIL	12.68	8.48	6.76	6.04

3.3.5 Thí nghiệm

Với mỗi cấu trúc cây T_Q , trong đó Q là số nút con tối đa tại mỗi nút, chúng tôi áp dụng các cách duyệt sau:

- Baseline: là phương pháp duyệt cơ bản: nút được chọn tiếp theo là nút có giá trị dự đoán cao nhất.
- ER-SHC: là phương pháp cải tiến của Zhu và cộng sự (CVIU2014).
- Traverse-MIL: là phương pháp được đề xuất trong nghiên cứu này.

Các thực nghiệm được tiến hành trên các tập dữ liệu Caltech-256, SUN-397, và ILSVRC2010-1K. Vector đặc trưng được sử dụng BOW-SIFT-LLC-SPM và phân chia dữ liệu tương tự như các thực nghiệm trước.

Các kết quả thực nghiệm trên tập dữ liệu ILSVRC2010-1K với các cấu hình cây khác nhau được liệt kê trong bảng 3.10 chứng minh độ chính xác phân loại của phương pháp được đề xuất luôn cao hơn so với các phương pháp khác.

3.4 Tổng kết chương

Chương này đã trình bày một cách chi tiết về các phương pháp xây dựng cây và phương pháp duyệt cây dựa trên mối quan hệ giữa của các nút. Các kết quả thực nghiệm trên các tập dữ liệu chuẩn như Caltech-256, SUN-397, ILSVRC2010-1K và ImageNet-10K đã chứng minh tính hiệu quả của các phương pháp đề xuất so với các phương pháp liên quan.

Chương 4 PHÂN LOẠI DỰA TRÊN BỘ PHÂN LỚP TIỀM ẨN

Tóm tắt

Chương này giới thiệu về cách tiếp cận dùng bộ phân lớp tiềm ẩn. Đây là một phương pháp mới được đề xuất trên cơ sở phỏng theo ý tưởng chính là giảm thiểu số lần gọi thực thi các bộ phân lớp, mà cách tiếp cận dùng cây phân cấp là một giải pháp. Trong cách tiếp cận này, phương pháp xấp xỉ ma trận bậc thấp được sử dụng để chuyển không gian ban đầu gồm C lớp sang không gian mới gồm L lớp tiềm ẩn, giá trị $L \ll C$. Các bộ phân lớp tiềm ẩn tương ứng được huấn luyện dựa trên phương pháp hồi quy. Kết quả phân loại của các bộ phân lớp tiềm ẩn trên tập dữ liệu kiểm tra được dùng để xác định kết quả phân loại cho chính tập dữ liệu kiểm tra đó trong không gian ban đầu. Nội dung của phương pháp này đã được công bố trong hội nghị quốc tế MMSP 2015 (oral presentation, ERA-B) [CT.5].

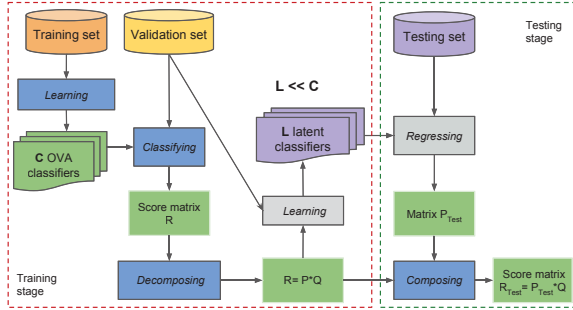
4.1 Giới thiệu

Trong chương này sẽ trình bày một cách tiếp cận mới được phát triển dựa trên phép xấp xỉ ma trận bậc thấp (low rank approximation) để phân tích mối liên quan và các đặc điểm tương đồng tồn tại một cách tiềm ẩn giữa các lớp. Ý tưởng chính là chỉ sử dụng một số lượng nhỏ L các bộ phân lớp tiềm ẩn để thực hiện phân lớp thay vì phải sử dụng sử dụng C bộ phân lớp nhị phân OvA.

4.2 Ý tưởng chính của phương pháp đề xuất

Để thực hiện phân loại cho một ảnh x , phương pháp thông thường (như phương pháp OvA) là áp dụng tất cả các bộ phân lớp lên ảnh x . Ảnh x sẽ được phân vào lớp nào có giá trị dự đoán cao nhất.

Nếu ta có M ảnh cần được phân vào một trong C lớp, thì ta sẽ có một ma trận $R_{M \times C}$ chứa các giá trị dự đoán của M ảnh. Dựa trên ma trận này, ta có thể xác định kết quả phân loại cho từng ảnh. Trong phương pháp được đề xuất, chúng ta chỉ áp dụng L bộ phân lớp, với $L \ll C$, để tạo ra một ma trận $R'_{M \times C}$ xấp xỉ tốt nhất với ma trận $R_{M \times C}$. Do chỉ có L bộ phân lớp được sử dụng để thực hiện phân loại,



Hình 4.2: Kiến trúc tổng quan của hệ thống thực hiện huấn luyện và phân lớp dựa trên các bộ phân lớp tiềm ẩn.

nên chi phí phân loại giảm đáng kể so với phương pháp OvA. Hình 4.2 minh họa kiến trúc tổng quan của hệ thống được đề xuất.

4.3 Xây dựng các bộ phân lớp tiềm ẩn

4.3.1 Mô tả phương pháp

Trước tiên, chúng ta đặt một số ký hiệu như sau:

- $Y = \{1, 2, \dots, C\}$ là tập nhãn của C lớp.
- $X = \mathbb{R}^D$ là không gian đặc trưng kích thước D chiều.
- \mathcal{V} là tập dữ liệu kiểm thử gồm có N ảnh, mỗi ảnh $x_i \in X$ được gán nhãn y_i . Ta có:

$$\mathcal{V} = \{(x_i, y_i) \mid x_i \in X, y_i \in Y, \cup_{i=1}^N y_i = Y\}.$$
- $R_{N \times C}$ là ma trận thể hiện mối quan hệ giữa N ảnh trong tập \mathcal{V} với C lớp. Chẳng hạn, mỗi phần tử $R_{i,j}$ là giá trị dự đoán của bộ phân lớp nhị phân thứ j trên ảnh x_i , giá trị này cho biết khả năng ảnh x_i được gán vào lớp thứ j .

Áp dụng phương pháp xấp xỉ ma trận bậc thấp để xấp xỉ ma trận $R_{N \times C}$ thành tích của hai ma trận $P_{N \times L}$ và $Q_{L \times C}$, trong đó L là giá trị bậc hay số lượng lớp tiềm ẩn mong muốn:

$$R_{N \times C} = P_{N \times L} Q_{L \times C} \quad (4.1)$$

Ta có thể xem:

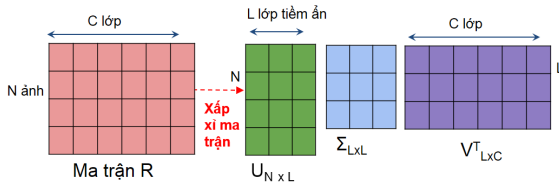
- Mỗi giá trị $P_{(i, k)}$ trong ma trận P có thể xem như là giá trị đoán của bộ phân lớp mới thứ k , được ký hiệu là $g_k(\cdot)$, trên vector đặc trưng x_i của ảnh thứ i .
- Mỗi giá trị $Q_{k, j}$ trong ma trận Q thể hiện mối quan hệ giữa lớp mới thứ k với lớp thứ j .

Ta có thể xem ma trận $R \in X \times Y$ được biến đổi thành tích của hai ma trận $P \in X \times \mathcal{L}$ và ma trận $Q \in \mathcal{L} \times Y$. Không gian \mathcal{L} được xem như một không gian tiềm ẩn (latent space), khi đó các lớp mới trong không gian này được gọi là các lớp tiềm ẩn (latent class). Mỗi giá trị $P_{i, k}$ là giá trị dự đoán của ảnh x_i bởi bộ phân lớp tiềm ẩn (latent classifier) tương ứng với lớp tiềm ẩn thứ k .

Để thực hiện xấp xỉ một cách hiệu quả, chúng tôi sử dụng phương pháp xấp xỉ ma trận SVD với số bậc L . Khi đó, ma trận R được xấp xỉ như sau:

$$R_{N \times C} = U \Sigma V^T \quad (4.2)$$

trong đó $U_{N \times L}$ và $V_{C \times L}$ là ma trận chứa các vector riêng (eigenvector) bên trái và bên phải, $\Sigma_{L \times L}$ là ma trận đường chéo chứa L giá trị kì dị lớn nhất (largest singular values) của R . Kết quả của phép xấp xỉ ma trận này có thể được mô tả trực quan qua hình vẽ 4.3.



Hình 4.3: Minh họa kết quả áp dụng phép xấp xỉ ma trận bậc thấp để xấp xỉ ma trận R . Mỗi phần tử $R_{i, j}$ thể hiện mối quan hệ giữa ảnh x_i với lớp thứ j , $j = 1, \dots, C$. Mỗi phần tử $P_{i, k}$ thể hiện mối quan hệ giữa ảnh x_i với lớp tiềm ẩn thứ k , $k = 1, \dots, L$.

Trong trường hợp này, ta đặt:

$$P = U_{N \times L} \quad (4.3)$$

và

$$Q = \Sigma_{L \times L} V_{C \times L}^T \quad (4.4)$$

Ma trận $P_{N \times L}$ được xem như một ma trận chứa giá trị kết quả của L bộ phân lớp tiềm ẩn khi áp dụng lên N ảnh ảnh của tập kiểm thử \mathcal{V} . Mỗi giá trị $P_{i,k} = g_k(x_i)$ là giá trị dự đoán của bộ phân lớp tiềm ẩn g_k trên ảnh x_i . Áp dụng phương pháp hồi quy (regression) để tìm các tham số của mô hình $g_k(\cdot)$ tương ứng (do đó bộ phân lớp tiềm ẩn $g_k(\cdot)$ còn được gọi là hàm hồi quy $g_k(\cdot)$).

4.3.2 Thuật toán huấn luyện

Quá trình thực hiện huấn luyện các bộ phân lớp tiềm ẩn được trình bày tóm tắt trong thuật toán 4.1.

Algorithm 4.1 $[G, Q] = \text{Training}(\mathcal{V}, R_{N \times C}, L)$: huấn luyện các bộ phân lớp tiềm ẩn

Input:

- 1: \mathcal{V} : Tập hợp các feature vector của N ảnh dùng để huấn luyện các bộ phân lớp;
- 2: $R_{N \times C}$: Ma trận chứa các giá trị dự đoán của C bộ phân lớp nhị phân trên tập \mathcal{V} ;
- 3: L : Số lượng bộ phân lớp tiềm ẩn mong muốn;

Output:

- 4: $G = g_1, \dots, g_L$: Tập hợp gồm L bộ phân lớp tiềm ẩn;
 - 5: Ma trận Q ;
 - 6: **Bước 1:** Xấp xỉ ma trận $R_{N \times C}$ bằng phép phân rã SVD: $R = U \Sigma V^T$. Chỉ giữ lại L trị riêng lớn nhất.
 - 7: **Bước 2:** Đặt $P = U_{N \times L}$ và $Q = \Sigma_{L \times L} V_{C \times L}^T$.
 - 8: **Bước 3:** Với mỗi cột thứ k của ma trận P và N ảnh của tập dữ liệu \mathcal{V} , áp dụng phương pháp hồi quy ta tìm được các tham số tương ứng của bộ phân lớp tiềm ẩn $g_k(\cdot)$.
-

4.3.3 Thuật toán phân loại

4.3.3.1 Mô tả thuật toán

Để thực hiện phân lớp cho một tập \mathcal{T} chứa M ảnh, chúng ta áp dụng các bộ phân lớp tiềm ẩn g_k trên M ảnh để tìm giá trị $P_{\mathcal{T}}$. Tiếp

theo ta tính được ma trận chứa các giá trị dự đoán cho M ảnh này dựa trên phương pháp kết hợp ma trận: $R_{\mathcal{T}} = P_{\mathcal{T}}Q$. Quá trình thực hiện phân lớp được trình bày tóm tắt trong thuật toán 4.2.

Algorithm 4.2 [$R_{\mathcal{T}} = Testing(G, Q, \mathcal{T})$]: phân lớp ảnh dựa trên các bộ phân lớp tiềm ẩn

Input:

- 1: $G = g_1, \dots, g_L$: Tập hợp gồm L bộ phân lớp tiềm ẩn;
- 2: Ma trận Q tính được trong quá trình học các bộ phân lớp tiềm ẩn;
- 3: \mathcal{T} : Tập hợp các vector đặc trưng của M ảnh cần được phân lớp;

Output:

- 4: Ma trận chứa các giá trị dự đoán M ảnh đối với C lớp: $R_{\mathcal{T}}$
 - 5: **Bước 1:** Áp dụng L bộ phân lớp lên tập \mathcal{T} , ta được ma trận $P_{\mathcal{T}} = [g_1(\mathcal{T}), \dots, g_L(\mathcal{T})]$.
 - 6: **Bước 2:** Tính giá trị của ma trận kết quả: $R_{\mathcal{T}} = P_{\mathcal{T}}Q$
-

4.3.3.2 Phân tích độ phức tạp của các thuật toán

Giả sử tập dữ liệu $T_{M \times D}$ cần được phân loại chứa M ảnh, mỗi ảnh được biểu diễn bằng một vector đặc trưng có kích thước D chiều.

- Để giảm độ phức tạp về tính toán khi thực hiện phân loại tập ảnh này bằng C bộ phân lớp nhị phân OvA, ta sử dụng các bộ phân lớp nhị phân tuyến tính. Vector trọng số của mô hình tương ứng với bộ phân lớp nhị phân tuyến tính OvA có kích thước $D \times 1$.

Gọi $W_{D \times C}$ là ma trận chứa C vector trọng số của C bộ phân lớp nhị phân tuyến tính OvA. Khi đó, ma trận kết quả phân loại có thể được xác định như sau:

$$R_{M \times C}^{OvA} = T_{M \times D} \cdot W_{D \times C} \quad (4.5)$$

Tổng số phép toán tích vô hướng (dot product) phải thực hiện để tạo ra ma trận dự đoán R^{OvA} là $\Theta^{OvA} = M * D * C$.

- Khi áp dụng thuật toán phân loại được đề xuất. Số phép toán tích vô hướng phải thực hiện để tạo ra ma trận dự đoán $P_{M \times L}$ là $M * D * L$.

Số phép toán tích vô hướng phải thực hiện để tạo ra ma trận dự đoán $R_{M \times C}^{Latent} = P_{M \times D} \cdot Q_{L \times C}$ là $M * L * C$.

Như vậy, tổng số phép toán tích vô hướng phải thực hiện để tạo ra ma trận dự đoán R^{Latent} sẽ là: $\Theta^{Latent} = M * D * L + M * L * C = M * (D + C) * L$. Khi số lượng bộ phân lớp tiềm ẩn $L = \frac{C}{k}$, số phép toán $\Theta^{Latent} = M * \frac{D+C}{k} * C$. Trường hợp xấu nhất tương ứng với giá trị $k = 1$, độ phức tạp Θ^{Latent} lớn hơn nhưng không đáng kể so với Θ^{OvA} . Với các giá trị $k > 1$, độ phức tạp của phương pháp đề xuất thấp hơn so với phương pháp OvA.

4.4 Thực nghiệm

4.4.1 Thiết lập thực nghiệm

Các thực nghiệm được tiến hành trên các tập dữ liệu chuẩn Caltech-256, SUN-397, ILSVRC2010-1K và ImageNet-10K. Hai loại đặc trưng được sử dụng để thực nghiệm tương tự như trong chương 3: đặc trưng BOW-SIFT-LLC-SPM và VGG-VERYDEEP-16. Sử dụng độ đo chính xác theo top-1 và top-5.

Ngoài độ chính xác của phương pháp phân loại, thời gian thực hiện phân loại cũng được xét. Đây là thời gian cần thiết để thực hiện phân lớp cho một tập ảnh kiểm tra.

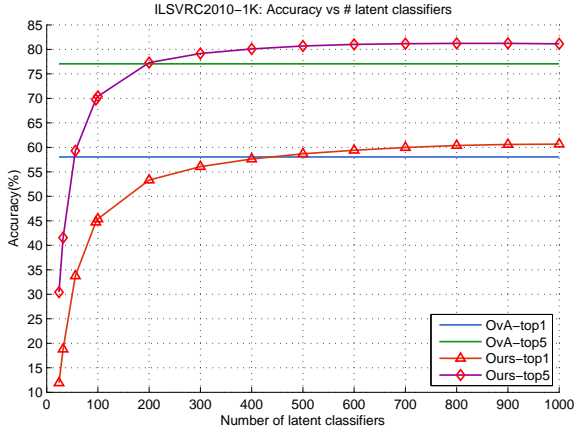
4.4.2 Kết quả thực nghiệm

4.4.2.1 Độ chính xác phân lớp

Trong bảng tóm tắt luận án, chúng tôi xin được trình bày kết quả trên tập dữ liệu ILSVRC2010-1K và đặc trưng VGG-VERYDEEP-16. Kết quả được minh họa trong hình 4.4.

Quan sát các kết quả thực nghiệm, chúng ta có một số nhận xét sau:

- Khi đánh giá theo độ chính xác top-1, *phương pháp đề xuất chỉ sử dụng không đến phân nửa số bộ phân lớp tiềm ẩn để đạt độ chính xác như sử dụng toàn bộ 1000 bộ phân lớp nhị phân OvA.*
- Khi đánh giá theo độ chính xác top-5, *phương pháp đề xuất chỉ dùng khoảng 1/5 số lượng bộ phân lớp tiềm ẩn là đạt được độ chính xác như dùng tất cả bộ phân lớp OvA.* Cụ thể: chỉ cần sử dụng 200 bộ phân lớp tiềm ẩn ta có thể đạt được độ chính xác tương đương với dùng 1000 bộ phân lớp nhị phân OvA.



Hình 4.4: Minh họa kết quả sử dụng đặc trưng VGG-VERYDEEP-16. Trong trường hợp độ chính xác top-1, chúng ta chỉ dùng khoảng phân nửa bộ phân lớp tiềm ẩn sẽ có độ chính xác tương đương với trường hợp dùng **1.000** bộ phân lớp nhị phân.

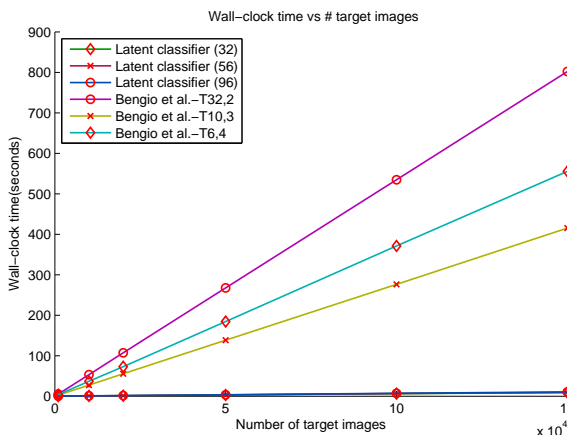
4.4.2.2 So sánh với phương pháp phân loại khác

Để đánh giá hiệu quả của phương pháp được đề xuất, chúng tôi so sánh với các phương pháp dùng cây phân cấp và phương pháp ECOC. Các kết quả thực nghiệm cho thấy: với cùng một chi phí phân lớp, độ chính xác của phương pháp được đề xuất đều cao hơn các phương pháp liên quan.

4.4.2.3 Phân tích sự ảnh hưởng của ma trận quan hệ

Ma trận quan hệ $R_{N \times C}$ thể hiện mối quan hệ giữa N ảnh trong tập \mathcal{V} với C lớp. Từ các kết quả thực nghiệm trên tập dữ liệu ILSVRC2010-1K, chúng ta có thể nhận thấy:

- *Độ chính xác của kết quả phân lớp bằng các bộ phân lớp tiềm ẩn phụ thuộc vào độ chính xác của ma trận R . Khi độ chính xác của R tăng thì độ chính xác phân lớp sử dụng các bộ phân lớp tiềm ẩn sẽ tăng.*
- *Số ảnh dùng để huấn luyện các bộ phân lớp tiềm ẩn không ảnh hưởng nhiều đến kết quả phân lớp. Sử dụng 30 ảnh/lớp hay 50 ảnh/lớp để huấn luyện thì độ chính xác phân lớp là tương đương nhau.*



Hình 4.10: So sánh thời gian (tính theo giây) thực hiện phân lớp của các phương pháp theo kích thước tập dữ liệu kiểm tra.

4.4.2.4 So sánh về thời gian thực hiện

Ngoài độ chính xác của phương pháp phân loại. Chúng tôi tiến hành so sánh về thời gian so với phương pháp dùng cây và phương pháp dùng các bộ phân lớp nhị phân OvA. Hình 4.10 minh họa sự thay đổi về thời gian theo số lượng ảnh test.

Từ các kết quả thực nghiệm ta có thể thấy: thời gian của phương pháp đề xuất rất nhỏ so với phương pháp khác. Lý do là vì trong phương pháp dùng cây, chúng ta phải duyệt tuần tự từng node để quyết định node nào sẽ được duyệt tiếp theo, vì thế chúng ta mất chi phí cho quá trình lựa chọn và chi phí cho quá trình chuyển tiếp giữa các node. Phương pháp OvA dùng tất cả các lớp nên có chi phí tính toán lớn hơn.

4.4.2.5 So sánh các phương pháp phân rã ma trận

Ngoài phương pháp SVD, một số phương pháp phân rã khác (Non-negative Matrix Factorization) đã được thực nghiệm như các phương pháp als, alsobs, mm, cjlin,.... Kết quả cho thấy phương pháp SVD cho kết quả phân loại tốt nhất và phù hợp với cách tiếp cận được đề xuất.

4.5 So sánh với các phương pháp phân loại dựa trên cây phân cấp đã đề xuất

Để so sánh hiệu quả của phương pháp phân loại dựa trên các bộ phân lớp tiềm ẩn so với các phương pháp được đề xuất trong phần 3.2.2 và phần 3.2.3, các thực nghiệm được tiến hành trên cùng tập dữ liệu và theo các tham số như trong phần 3.2.4. Số lượng bộ phân lớp tiềm ẩn được sử dụng tương đương với số lượng bộ phân lớp được sử dụng trong cấu trúc cây. Chẳng hạn, với cấu trúc cây $T_{10,3}$ có giá trị $S_{te} = 33.33$, ta cần $L = \frac{1000}{33.33} \approx 30$ bộ phân lớp tiềm ẩn. Bảng 4.5 liệt kê các kết quả thực nghiệm.

Bảng 4.5: So sánh hiệu quả của các phương pháp đề xuất trên tập dữ liệu ILSVRC2010-1K với đặc trưng VGG-VERYDEEP-16.

Phương pháp	$T_{32,2}(L = 60)$		$T_{10,3}(L = 30)$		$T_{6,4}(L = 24)$	
	<i>Acc</i>	<i>S_{te}</i>	<i>Acc</i>	<i>S_{te}</i>	<i>Acc</i>	<i>S_{te}</i>
Bengio et al.	53.21	16.01	44.27	32.59	40.06	39.64
Liu et al.	56.91	15.84	54.27	28.21	52.71	37.90
BLTree-SMK- <i>kchi2</i>	57.77	15.77	54.40	33.33	51.97	43.10
BLTree-SMK- <i>kinters</i>	57.82	15.74	54.26	33.33	52.75	43.42
BLTree-SMK- <i>kjs</i>	57.87	15.74	54.38	33.33	52.70	43.30
BLTree-AM	58.13	15.76	54.84	33.33	52.91	42.46
Latent classifiers	33.54	16.67	16.91	33.33	11.80	41.67
OvA	55.73	1.0				

Bảng kết quả 4.5 đã chứng minh với số lượng bộ phân lớp tương đương, phương pháp sử dụng cây phân cấp có độ chính xác tốt hơn. Nếu xét về thời gian thực hiện thì phương pháp dùng các lớp tiềm ẩn nhanh hơn so với phương pháp dùng cây.

4.6 Tổng kết chương

Chương này đã trình bày cách tiếp cận dựa trên các bộ phân lớp tiềm ẩn để tạo ra ma trận dự đoán với chi phí thực hiện thấp hơn; độ chính xác phân lớp tốt hơn.

Chương 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

5.1.1 Những kết quả đạt được

Trong luận án này, nghiên cứu sinh đã nghiên cứu và phát triển một số thuật toán hiệu quả như để giải quyết một số vấn đề thách thức của bài toán phân loại ảnh với số lượng lớn các lớp. Cụ thể như sau:

- a. Phát triển phương pháp xây dựng cấu trúc cây phân cấp cân bằng dựa trên tất cả các ảnh và ảnh trung bình.
- b. Phát triển phương pháp xây dựng cấu trúc cây phân cấp dựa trên sự tương đồng giữa các lớp.
- c. Phát triển phương pháp thực hiện phân loại dựa trên thông tin về mối quan hệ giữa các nút.
- d. Đề xuất phương pháp thực hiện phân loại dựa trên các bộ phân lớp tiềm ẩn. Phương pháp này không chỉ giảm chi phí thực hiện mà còn cho phép chúng ta có thể chọn số lượng bộ phân lớp tiềm ẩn tương ứng để có thể sử dụng tốt nhất tài nguyên của hệ thống nhằm đạt được một độ chính xác tốt nhất.

Các kết quả nghiên cứu trên đã được công bố trên hai tạp chí chuyên ngành ([CT.1] và [CT.2]) và các hội thảo chuyên ngành được phản biện độc lập ([CT.3], [CT.4],[CT.5],[CT.6]).

5.1.2 Những ưu điểm và hạn chế của các phương pháp đề xuất

- Các phương pháp đề xuất trong hướng tiếp cận dựa trên cây phân cấp có những ưu điểm chính sau: Do cấu trúc cây được đảm bảo tính cân bằng nên chi phí phân loại thấp hơn; độ chính xác phân loại cao hơn; phù hợp với các hệ thống phân loại có số lượng lớp lớn; Tuy nhiên, hướng tiếp cận này có một số hạn chế sau:chưa xét đến yếu tố mất cân bằng của các lớp trong tập dữ liệu kiểm tra; khó xác định một cấu hình cây để có thể khai thác tối ưu tài nguyên hệ thống.

- Phương pháp dựa trên các bộ phân lớp tiềm ẩn có một số ưu điểm chính sau: độ chính xác phân lớp cao hơn; chi phí phân lớp thấp hơn so với phương pháp OvA; có thể điều chỉnh số lượng bộ phân lớp tiềm ẩn để đạt được độ chính xác tốt nhất trên tài nguyên hệ thống; có thể thực hiện song song hóa dễ dàng; thích hợp cho bài toán phân loại đồng thời cho một tập kiểm tra có kích thước lớn. Mặc dù có nhiều ưu điểm như trên, phương pháp này có một số hạn chế sau: số lượng bộ phân lớp tiềm ẩn càng nhỏ thì độ chính xác càng giảm; phải huấn luyện các bộ phân lớp nhị phân để xác định ma trận quan hệ R .

5.2 Hướng phát triển

Trong luận án này, nghiên cứu sinh đã phát triển một số phương pháp phân loại hiệu quả. Tuy nhiên, chúng vẫn còn một số hạn chế và cần tiếp tục cải tiến và hoàn thiện hơn nữa các phương pháp khác đã được phát triển trong luận án như: phát triển các phương pháp xây dựng cây phân cấp trên các tập dữ liệu mất cân bằng; phát triển các phương pháp xác định các bộ phân lớp tiềm ẩn hiệu quả hơn; khai thác các mối quan hệ tiềm ẩn trong các lớp cho các bài toán khác; cải tiến và ứng dụng các phương pháp được đề xuất vào các bài toán khác.

DANH MỤC CÔNG BỐ KHOA HỌC

Tạp chí chuyên ngành

- [CT.1] Tien-Dung Mai. Using sum match kernel with balanced label tree for large-scale image classification. *Journal of Computer Science and Cybernetics*, vol.32, No.2, pages 133–152, 2016.
- [CT.2] Tien-Dung Mai, Thanh Duc Ngo, Duy-Dinh Le, Duc Anh Duong, Kiem Hoang, and Shin'ichi Satoh. Efficient Large-Scale Multi-Class Image Classification by Learning Balanced Trees. *Journal of Computer Vision and Image Understanding*, vol.156, pages 151–161, 2017.

Hội thảo chuyên ngành

- [CT.3] Tien-Dung Mai, Thanh Duc Ngo, Duy-Dinh Le, Duc Anh Duong, Kiem Hoang, and Shin'ichi Satoh. Learning balanced trees for large scale image classification. In ICIAP (2) 2015, pages 3-13, 7-11 September 2015, Genova, Italy.
- [CT.4] Tien-Dung Mai, and Kiem Hoang. Label tree based image classification using sum match kernel. In ATC 2015, pages 468 - 472, 14-16 October 2015, Ho Chi Minh City, Vietnam.
- [CT.5] Tien-Dung Mai, Thanh Duc Ngo, Duy-Dinh Le, Duc Anh Duong, Kiem Hoang, and Shin'ichi Satoh. Large scale multi-class classification using latent classifiers. In MMSP' 15, pages 1-6, 19 - 21 October 2015, Xiamen, China.
- [CT.6] Tien-Dung Mai, Thanh Duc Ngo, Duy-Dinh Le, Duc Anh Duong, Kiem Hoang, and Shin'ichi Satoh. Using node relationships for hierarchical classification. In ICIP 2016, pages 514-518, 25-28 September 2016, Phoenix, Arizona, USA.