

ĐẶC TRƯNG DENSE SIFT VÀ THUẬT TOÁN LOCAL NAIVE BAYES NEAREST NEIGHBOR TRONG NHẬN DẠNG MẶT NGƯỜI

DENSE SIFT FEATURE AND LOCAL NAIVE BAYES NEAREST NEIGHBOR FOR FACE RECOGNITION

Châu Ngân Khánh¹, Đoàn Thanh Nghị²

Tóm tắt – Công nghệ nhận dạng mặt người đã được ứng dụng rộng rãi trong đời sống. Chúng ta đã có nhiều thuật toán được nghiên cứu để giải quyết bài toán này. Trong bài báo này, chúng tôi trình bày phương pháp mới, kết hợp Haar Like Feature - Cascade of Boosted Classifiers, Dense Scale-Invariant Feature Transform (DSIFT), thuật toán Local Naive Bayes Nearest Neighbor (LNBNN) để nhận dạng mặt người. Đầu tiên, rút trích ảnh khuôn mặt bằng thuật toán AdaBoost với mô hình Cascade trên đặc trưng Haar Like, tính các DSIFT trên ảnh đã rút trích. Sau đó, sử dụng LNBNN để nhận dạng. Kết quả thử nghiệm trên các tập dữ liệu kiểm chuẩn cho thấy phương pháp này đạt kết quả tốt hơn các phương pháp khác, độ chính xác của LNBNN là 99.74%

Từ khóa: đặc trưng Haar Like, thuật toán AdaBoost, mô hình phân tầng Cascade, đặc trưng SIFT, nhận dạng khuôn mặt, thuật toán Local Naive Bayes Nearest Neighbor.

Abstract – Human face recognition is a technology which is widely used in life. There have been much effort on developing face recognition algorithms. In this paper, we present a new methodology that combines Haar Like Features - Cascade of Boosted Classifiers, Dense Scale-Invariant Feature Transform (DSIFT), Local Naive Bayes Nearest Neighbor (LNBNN) algorithm for the recognition of human face. We use Haar Like Features and the combination

of AdaBoost algorithm and Cascade stratified model to detect and extract the face image, the DSIFT descriptors of the image are computed only for the aligned and cropped face image. Then, we apply the LNBNN algorithms for object recognition. Numerical testing on several benchmark datasets using our proposed method for face recognition gives the better results than other methods. The accuracies obtained by LNBNN method is 99.74 %.

Keywords: haar Like Features, AdaBoost Algorithm, Cascade of Boosted Classifiers, DSIFT, Face recognition, Local Naive Bayes Nearest Neighbor.

I. ĐẶT VẤN ĐỀ

Cùng với sự phát triển công nghệ cao, nhận dạng mặt người ngày càng được ứng dụng rộng rãi, nhiều hệ thống nhận mặt người ra đời và giữ vai trò quan trọng trong đời sống con người. Hệ thống nhận dạng mặt người cơ bản gồm có các thành phần: thiết bị thu thập hình ảnh, cơ sở dữ liệu các hình ảnh, các thuật toán xử lý nhận dạng và thiết bị hiển thị kết quả nhận dạng. Trong đó, các thuật toán xử lý nhận dạng đóng vai trò quyết định tốc độ và độ chính xác của hệ thống nhận dạng mặt người.

Các nhà khoa học đã đề xuất nhiều phương pháp khác nhau nhằm giải quyết bài toán nhận dạng mặt người. Trong đó, hướng tiếp cận dựa trên các đặc trưng cục bộ không đổi SIFT của Lowe [1] được quan tâm nhiều. DSIFT là một phiên bản tính toán nhanh các đặc trưng SIFT với mật độ dày, do Vedaldi et al. phát triển và được chứng minh cho kết quả tốt với các bài toán phân lớp ảnh [2]. Trong khi đó, LNBNN được đề xuất

^{1,2}Khoa Công nghệ Thông tin, Trường Đại học An Giang
Email: cnkhanh@agu.edu.vn

Ngày nhận bài: 04/8/2017; Ngày nhận kết quả bình
duyet: 13/11/2017; Ngày chấp nhận đăng: 29/12/2017

để cải thiện đáng kể độ chính xác cũng như thời gian thực hiện so với thuật toán NBNN trên các tập dữ liệu lớn. Trong bài báo này, chúng tôi đề xuất sử dụng các đặc trưng Haar Like với thuật toán AdaBoost và mô hình phân tầng Cascade để định vị khuôn mặt. Sau đó, rút trích ảnh khuôn mặt người vừa phát hiện ta sẽ thu được ảnh đối tượng. Tính đặc trưng DSIFT của ảnh đối tượng. Đặc trưng DSIFT đã được chứng minh cho kết quả cao trong việc phân lớp ảnh [2]. Cuối cùng, thuật toán phân lớp LNBNN [3] được sử dụng để nhận dạng ảnh, đây là thuật toán phân lớp ảnh đơn giản và có độ chính xác cao khi phân lớp. Mô hình hệ thống được mô tả như Hình 1.

Phần tiếp theo của bài viết được trình bày như sau: Phần 2 trình bày tổng quan các nghiên cứu liên quan, phần 3 giới thiệu ngắn gọn về thuật toán phát hiện khuôn mặt Haar Like – Cascade of Boosted Classifiers, biểu diễn ảnh bằng các đặc trưng DSIFT được trình bày trong phần 4, phần 5 trình bày phương pháp định danh với LNBNN, phần 6 trình bày các kết quả thực nghiệm trước khi vào phần kết luận và hướng phát triển.

II. TỔNG QUAN NGHIÊN CỨU

Các nghiên cứu trước đây, Kirby và Sirovich [4]; Turk và Pentland [5] nhận dạng khuôn mặt bằng phương pháp PCA và thuật toán eigenfaces. Trần Phước Long và Nguyễn Văn Lượng [6], dò tìm khuôn mặt trong ảnh, kết hợp với phương pháp PCA và DCT để rút ra các đặc trưng làm đầu vào cho bộ nhận dạng SVM và mô hình HMM. Lu Boun Vinh và Hoàng Phương Anh [7] thực hiện dò tìm khuôn mặt kết hợp với thuật toán FSVM để tiến hành nhận dạng mặt người. Năm 2006, Zuo và các cộng sự [8] đã kết hợp phương pháp PCA hai chiều với LDA để nhận dạng mặt người; Chen và cộng sự [9] kết hợp DCT và HRBF; Aly [10] sử dụng các đặc trưng SIFT để nhận dạng mặt người. Năm 2011, Chennamma và các cộng sự [11] thực hiện nhận dạng mặt người trên tập ảnh đã được chọn lọc trước bằng thuật toán đối sánh các đặc trưng SIFT. Năm 2012, Sharif và các cộng sự [12] kết hợp thuật toán LoG và DCT; Suhas và các cộng sự [13] áp dụng PCA kết hợp với DCT theo hướng tiếp cận toàn diện; Kumar và Padmavati [14] đã đề xuất các cách tính khoảng cách khác nhau giữa các đặc trưng

SIFT nhằm cải thiện thuật toán SIFT. Năm 2013, Sunil [15] kết hợp PCA với DCT; Bouzalmat và các cộng sự [4] áp dụng thuật toán SVM dựa trên LDA, để nhận dạng khuôn mặt người trên ảnh. Châu Ngân Khánh và Đoàn Thanh Nghị [16], kết hợp đặc trưng SIFT và thuật toán NBNN để nhận dạng mặt người. Năm 2015, Ladisla vLenc và Pavel Král [17] xây dựng hệ thống nhận dạng mặt người tự động dựa trên các đặc trưng SIFT. Năm 2016, Ehsan Sadeghipour và Nasrollah Sahragard [18] thực hiện cải tiến thuật toán SIFT để nhận dạng mặt người.

Các nhà nghiên cứu đã đề ra nhiều phương pháp khác nhau để giải quyết bài toán nhận dạng mặt người. Tuy nhiên, các giải pháp đã đề ra vẫn chưa đạt được kết quả tối ưu khi nhận dạng về độ chính xác cũng như thời gian thực hiện nhận dạng. Phương pháp do chúng tôi đề xuất nhằm tối ưu độ chính xác cũng như cải thiện thời gian thực hiện nhận dạng.

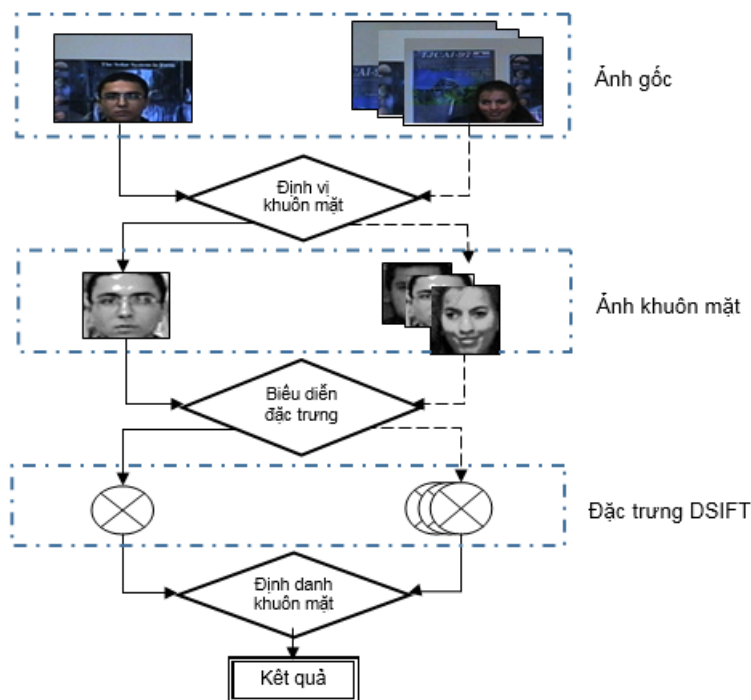
III. ĐỊNH VỊ KHUÔN MẶT

A. Đặc trưng Haar Like

Đặc trưng Haar Like được tạo thành do việc kết hợp các hình chữ nhật đen, trắng với nhau theo một trật tự, một kích thước nào đó. Để phát hiện khuôn mặt, các đặc trưng Haar Like cơ bản trên được mở rộng [19] thành nhóm các đặc trưng cạnh, đường và tâm.

B. Thuật toán AdaBoost

AdaBoost [19] là một bộ phân loại mạnh phi tuyến phức tạp, hoạt động dựa trên nguyên tắc kết hợp tuyến tính các bộ phân loại yếu để tạo nên một bộ phân loại mạnh. AdaBoost sử dụng trọng số để đánh dấu các mẫu khó nhận dạng. Trong quá trình huấn luyện, cứ mỗi bộ phân loại yếu được xây dựng thì thuật toán sẽ tiến hành tăng trọng số của các mẫu nhận dạng sai và giảm trọng số của các mẫu được nhận dạng đúng để chuẩn bị cho việc xây dựng bộ phân loại tiếp theo. Bằng cách này, bộ phân loại sau có thể tập trung vào các mẫu mà bộ phân loại trước nó làm chưa tốt. Cuối cùng, các bộ phân loại yếu sẽ được kết hợp lại tùy theo mức độ tốt của chúng để tạo nên một bộ phân loại mạnh.



Hình 1: Hệ thống nhận dạng mặt người

C. Mô hình phân tầng Cascade

Cascade of Boosted Classifiers là mô hình phân tầng với mỗi tầng là một mô hình AdaBoost sử dụng bộ phân lớp yếu là cây quyết định với các đặc trưng Haar-Like. Một mẫu để được phân loại là đối tượng thì nó cần phải đi qua hết tất cả các tầng. Các tầng sau được huấn luyện bằng những mẫu âm (negative - không phải mặt người) mà tầng trước nó nhận dạng sai, tức là nó sẽ tập trung học từ các mẫu khó hơn, do đó sự kết hợp các tầng AdaBoost này lại sẽ giúp bộ phân loại giảm thiểu nhận dạng lầm. Cấu trúc này cải tiến đáng kể thời gian xử lý và vẫn duy trì được hiệu quả phát hiện khuôn mặt.

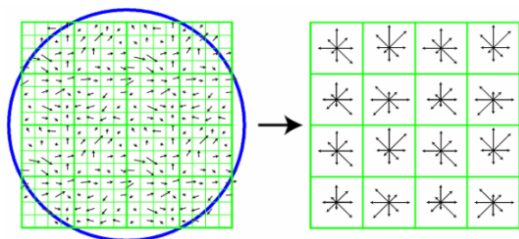
IV. BIỂU DIỄN ĐẶC TRƯNG KHÔNG ĐỔI

Rút trích các đặc trưng ảnh là một bước quan trọng trong nhận dạng ảnh. Bước này giúp biểu diễn ảnh bằng các đặc trưng quan trọng mà giải thuật có thể thực hiện nhận dạng ảnh từ các đặc trưng này. Hai tiếp cận chính cho biểu diễn ảnh hiện nay là: sử dụng nét đặc trưng toàn cục (global features) và nét đặc trưng cục bộ (local features). Trong đó, hướng tiếp cận thứ hai, đặc

trung cục bộ SIFT [1] đạt được hiệu quả cao trong nhận dạng và tìm kiếm ảnh. Năm 2007, Bosch và các cộng sự đã đề nghị đặc trưng DSIFT, một mở rộng của SIFT, cho việc nhận dạng ảnh và đạt được kết quả rất cao trong thực nghiệm [2]. Đặc trưng DSIFT được rút trích bằng cách tính SIFT trên một lưới dày đặc các vị trí trong ảnh với một tỉ lệ và hướng cố định. Chính vì lí do đó, chúng tôi đề xuất sử dụng các nét đặc trưng DSIFT để biểu diễn ảnh phục vụ cho quá trình nhận dạng.

Véc-tơ mô tả SIFT được đánh giá rất cao bởi các chuyên gia trong việc biểu diễn các vùng xung quanh điểm đặc trưng bởi vì nó không đổi đối với những biến đổi tỉ lệ, tịnh tiến, phép quay, và không đổi một phần đối với những thay đổi về góc nhìn, đồng thời nó cũng rất mạnh với những thay đổi về độ sáng, sự che khuất và nhiễu. Việc rút trích đặc trưng SIFT của một ảnh được mô tả tóm tắt như sau: Ảnh được đưa về dạng mức xám. Các điểm đặc trưng được tính trên ảnh này bằng cách sử dụng các giải thuật phát hiện điểm đặc trưng cục bộ (local feature detector) như là Harris-Affine, Hessian-Affine. Những điểm đặc

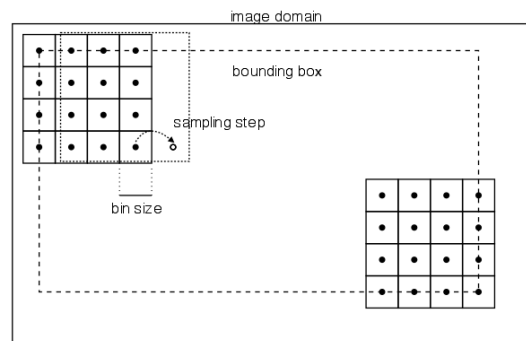
trung này có thể là cực trị cục bộ của phép toán DoG (Difference of Gaussian) hoặc là cực đại của phép toán LoG (Laplace of Gaussian). Sau đó, vùng xung quanh các điểm đặc trưng được xác định và mô tả bằng các véc-tơ mô tả cục bộ (Hình 2). Dựa trên SIFT, DSIFT đưa ra một số giả định mới: (a) Vị trí của một đặc trưng không xuất phát từ gradient của mỗi điểm ảnh, nhưng từ một vị trí đã được dự kiến trước; (b) Quy mô của mỗi đặc trưng cũng được dự tính trước; (c) Nhiều đặc trưng có hướng là 0. Với giả định này, DSIFT có thể tính được nhiều đặc trưng hơn với thời gian ngắn hơn thuật toán SIFT. Thay vì chụp ảnh gốc với một cửa sổ của hàm Gaussian (thực hiện đối với giải thuật SIFT) thuật toán DSIFT sẽ dùng một cửa sổ phẳng (Hình 3). Trong trường hợp này, biên độ gradient sẽ được tích lũy bằng cách sử dụng phép nội suy song tuyến, nhưng thay vì cập nhật lại trọng số bởi hàm Gaussian, tất cả sẽ được đánh trọng số như nhau. Sau khi tính được biên độ gradient trong không gian nhị phân (a spatial bin), tất cả sẽ được cập nhật trọng số bằng giá trị trung bình của cửa sổ Gaussian. Điều này cải thiện đáng kể tốc độ thực hiện mà không làm giảm hiệu suất của thuật toán hoặc hiệu suất giảm ít. Đặc trưng được lấy mẫu theo cách mà tâm của các không gian nhị phân là một tọa độ nguyên trong phạm vi của ảnh. Ví dụ, trong Hình 3, không gian nhị phân phía trên bên trái của mô tả phía trên bên trái sẽ tập trung vào tọa độ (0, 0).



Hình 2: Đặc trưng SIFT được tính toán từ vùng xung quanh điểm đặc biệt (vòng tròn): gradient của ảnh (trái), véc-tơ mô tả (phải)
(Nguồn: Lowe [1])

V. PHƯƠNG PHÁP ĐỊNH DANH

Trong bài báo này, chúng tôi đề xuất sử dụng thuật toán LNBN để định danh đối tượng.



Hình 3: Mô tả hình học của DSIFT
(Nguồn: Vedaldi et al. [20])

NBNN được đề xuất bởi Boiman [21]. Đây là phương pháp phân loại ảnh bằng cách tính khoảng cách từ “ảnh đến các lớp” (image-to-classes). Do đó, nó cần phải duy trì nhiều cấu trúc tìm kiếm riêng biệt cho các lớp. Thuật toán LNBN, được đề nghị bởi McCann và cộng sự [3], là một sự cải tiến của NBNN. Trong đó, việc tìm kiếm nhãn của ảnh truy vấn được thực hiện chỉ dựa vào một cấu trúc tìm kiếm với dữ liệu được xây dựng từ toàn bộ tập ảnh, vì vậy hiệu quả hơn thuật toán NBNN.

A. Thuật toán Naive Bayes Nearest Neighbor

Để xác định phân lớp của một ảnh truy vấn Q , ta cần tìm phân lớp C sao cho giá trị xác suất hậu nghiệm là cực đại (Maximum a posteriori), giá trị này được kí hiệu là \hat{C} .

$$\hat{C} = \underset{c}{\operatorname{argmax}} p(C|Q) \quad (1)$$

Với giả thiết xác suất phân lớp của dữ liệu quan sát được $p(Q)$ là như nhau đối với tất cả các phân lớp C , tất cả các phân lớp C đều có giá trị xác suất tiên nghiệm là như nhau. Gọi d_1, d_2, \dots, d_n là mô tả của các thành phần đặc trưng trong ảnh truy vấn Q . Giả sử các biến d_1, d_2, \dots, d_n là độc lập về xác suất, ta có:

$$\hat{C} = \underset{c}{\operatorname{argmax}} p(C|Q) = \underset{c}{\operatorname{argmax}} \prod_{i=1}^n p(d_i|C) \quad (2)$$

Tính logarithm của các giá trị xác suất p trong công thức (2). Áp dụng cửa sổ Parzen (Parzen window), ta có thể ước lượng xấp xỉ mật độ xác suất $p(d_i|C)$ với hàm nhân là K .

Thuật toán NBNN chuyển công thức (2) (sau khi đã tính logarithm) đến điểm tới hạn khi chỉ sử dụng một láng giềng gần nhất $NN_C(d_i)$:

$$\hat{p}_1(d_i|C) = \frac{1}{L} K(d_i - NN_C(d_i)) \quad (3)$$

Với L là tổng số các mô tả của các đặc trưng trong tập huấn luyện của lớp C .

Chọn hàm nhân Gaussian cho K và thay thế công thức (3) vào công thức (2) (sau khi đã tính logarithm), ta được:

$$\hat{C} = \underset{c}{\operatorname{argmax}} \left[\sum_{i=1}^n \log \frac{1}{L} e^{-\frac{1}{2\sigma^2} \|d_i - NN_c(d_i)\|^2} \right] \quad (4)$$

$$= \underset{c}{\operatorname{argmax}} \left[\sum_{i=1}^n \|d_i - NN_C(d_i)\|^2 \right] \quad (5)$$

Sau đó, McCann và cộng sự [3] đã đề nghị điều chỉnh việc cập nhật của NBNN sang việc cập nhật logarithm của tỉ lệ xác suất hậu nghiệm (log-odds) cho mỗi lớp. Trong đó, chỉ những cập nhật có giá trị dương cho một lớp là được quan tâm. Như vậy, tầm ảnh hưởng của mỗi đặc trưng trong ảnh truy vấn Q có thể được xem như một cập nhật của log-odds. Đặt C là một lớp nào đó, \bar{C} là tập tất cả các lớp còn lại, ta có.

$$\hat{C} = \underset{c}{\operatorname{argmax}} \left[\sum_{i=1}^n \frac{P(d_i|C)}{P(d_i|\bar{C})} + \log \frac{P(C)}{P(\bar{C})} \right] \quad (6)$$

Nếu xác suất tiên nghiệm là như nhau đối với các lớp thì ta có thể giản lược tỉ lệ xác suất tiên nghiệm. Trong khi đó, lượng tăng có thể được tính đơn giản nếu ta thay $P(d_i|C) \propto e^{-\|d_i - NN_C(d_i)\|^2}$. Điều này cho phép chúng ta chỉ cần sử dụng các cập nhật log-odds có ảnh hưởng lớn và nguyên dương. Thí nghiệm trong bài báo của McCann [3] đã chứng minh rằng việc điều chỉnh này không làm ảnh hưởng đến độ chính xác toàn cục khi phân lớp.

B. Thuật toán Naive Bayes Nearest Neighbor (NBNN)

Thuật toán phân lớp NBNN được thiết kế dựa theo công thức (6), các bước thực hiện như sau: trước hết, tính tất cả các mô tả d_1, d_2, \dots, d_n của ảnh truy vấn Q . Tiếp đến, cần xác định láng giềng gần nhất của d_i trong toàn bộ các mô tả của

tất cả các phân lớp C , được kí hiệu là $NN_C(d_i)$. Nhân của ảnh Q được gán cho phân lớp C nào có tổng khoảng cách Euclid từ d_i đến $NN_C(d_i)$ là nhỏ nhất. Thuật toán NBNN được tóm tắt như sau:

1. Tính toán tất cả các mô tả d_1, d_2, \dots, d_n của ảnh truy vấn Q và các mô tả của các ảnh có trong các phân lớp C của tập dữ liệu phân lớp.

2. $\forall d_i \forall C$ tìm láng giềng gần nhất của d_i trong C : $NN_C(d_i)$.

$$3. \hat{C} = \underset{c}{\operatorname{argmin}} \sum_{i=1}^n \|d_i - NN_C(d_i)\|^2$$

4. \hat{C} là kết quả trả về của thuật toán phân lớp.

C. Thuật toán Local Naive Bayes Nearest Neighbor (LNBNN)

LNBNN là thuật toán cải tiến của NBNN, giúp tăng tốc độ, cải thiện độ chính xác của thuật toán phân lớp và vì vậy làm tăng khả năng mở rộng đến số lượng lớn các phân lớp của tập dữ liệu đối tượng. McCann và cộng sự [3] đã chỉ ra rằng chỉ các lớp trong vùng lân cận cục bộ của một véc-tơ mô tả góp phần đáng kể và tin cậy vào việc ước tính xác suất hậu nghiệm. Thay vì duy trì một cấu trúc tìm kiếm riêng biệt cho mỗi phân lớp, tác giả đã kết hợp tất cả các dữ liệu đối tượng thành một cấu trúc tìm kiếm, cho phép xác định nhanh chóng vùng lân cận của một véc-tơ mô tả. Thuật toán LNBNN được đề xuất để tận dụng ưu điểm đã được đề cập ở công thức (6) với ảnh truy vấn Q và k láng giềng, các bước thực hiện như sau:

1. Tính toán tất cả các mô tả d_1, d_2, \dots, d_n của ảnh truy vấn Q và các mô tả của các ảnh có trong tập dữ liệu phân lớp.

2. Với mọi d_i , tìm $\{p_1, p_2, \dots, p_{k+1}\} \leftarrow NN(d_i, k+1)$ là $(k+1)$ láng giềng gần nhất của d_i .

a. Tính $\operatorname{dist}_B \leftarrow \|d_i - p_{k+1}\|_2$

b. Với mọi phân lớp C có chứa k láng giềng gần nhất, tính:

i. $\operatorname{dist}_C = \min_{p_j | \operatorname{Class}(p_j)=C} \|d_i - p_j\|^2$, với $\operatorname{Class}(p_j)$ là phân lớp chứa mô tả p_j .

ii. $\operatorname{totals}[C] \leftarrow \operatorname{totals}[C] + \operatorname{dist}_C - \operatorname{dist}_B$

3. $\hat{C} = \underset{c}{\operatorname{argmin}} \operatorname{totals}[C]$

4. \hat{C} là kết quả trả về của thuật toán phân lớp.

VI. THỰC NGHIỆM

Trong phần này, chúng tôi tiến hành đánh giá hiệu năng của hệ thống nhận dạng mặt người như đã đề xuất (sử dụng thuật toán Haar Like Features – Cascade of Boosted Classifiers, đặc trưng DSIFT và LNBNN). Hệ thống nhận dạng được cài đặt bằng ngôn ngữ lập trình C/C++, sử dụng thư viện OpenCV, VLFeat của Vedaldi, thư viện lập trình C++ (Boost C++), thư viện FLANN và một số thư viện của Google (Google Test, Google Command Line Flag, Google Logging, Google Protocol Buffers), trên một máy tính chạy hệ điều hành Linux với bản phân phối Fedora. Bước phát hiện mặt người được thực hiện thông qua việc huấn luyện mô hình phân tầng với mỗi tầng là một mô hình AdaBoost sử dụng bộ phân lớp yếu là cây quyết định với các đặc trưng Haar-Like trên tập ảnh (mặt người và không phải mặt người). Bước biểu diễn ảnh được thực hiện thông qua việc sử dụng module DSIFT được cung cấp bởi thư viện VLFeat.

Phương pháp của chúng tôi được đánh giá trên các tập dữ liệu kiểm chuẩn AT&T (ORL) được giới thiệu bởi AT&T Laboratories Cambridge (1994); Face94, Face95, Face96 được giới thiệu bởi Spacek (2007). Đầu tiên, chương trình sử dụng mô hình phân tầng đã huấn luyện để phát hiện mặt người, rút trích khuôn mặt. Sau đó, chúng tôi sử dụng lớp VL_DSIFT từ thư viện VLFeat để rút trích các đặc trưng DSIFT của tất cả các ảnh khuôn mặt và lưu vào một cơ sở dữ liệu theo các nhãn lớp. Kế đến, chúng tôi xây dựng cây chỉ mục được hỗ trợ bởi thư viện FLANN cho tập các đặc trưng nhằm giúp quá trình tìm kiếm các láng giềng được thực hiện một cách nhanh chóng. Cuối cùng, chúng tôi áp dụng giải thuật LNBNN để định danh khuôn mặt. Nghi thức kiểm tra trong thực nghiệm của chúng tôi là repeated hold-out [22], lấy ngẫu nhiên 2/3 tập dữ liệu làm tập học (hay cơ sở dữ liệu đối tượng), 1/3 tập dữ liệu còn lại làm tập kiểm tra. Chúng tôi thực hiện việc kiểm thử năm lần, sau đó tính trung bình cộng để xác định giá trị lỗi tổng thể. Tiếp theo, chúng tôi so sánh các kết quả thu được từ việc sử dụng giải thuật NBNN và LNBNN. Việc rút trích được khuôn mặt người từ ảnh gốc đã giảm lược khá nhiều số lượng DSIFT của đối tượng, nhờ vậy hệ thống đã tăng tốc đáng

kể quá trình nhận dạng và đạt được độ chính xác cao hơn. Nếu ảnh có kích thước 500×500 pixels thì có khoảng 40000 DSIFT. Nhưng khi trích xuất được khuôn mặt người trong ảnh, số lượng DSIFT trung bình chỉ còn khoảng 2000. Mặc dù số lượng DSIFT giảm đi nhưng độ chính xác của thuật toán không những không giảm mà còn tăng lên, lí do là vì các đặc trưng không hữu ích hoặc làm ảnh hưởng xấu đến kết quả nhận dạng đối tượng đã được loại bỏ. Thực nghiệm cho thấy số lượng DSIFT không hữu ích là lớn hơn rất nhiều so với DSIFT hữu ích trong nhận dạng. Để thấy được ưu điểm của đặc trưng DSIFT so với SIFT, chúng tôi tiến hành so sánh số lượng đặc trưng tìm được trên các tập dữ liệu ảnh (Bảng 1), đồng thời thực hiện đánh giá hiệu quả của các đặc trưng đối với hệ thống nhận dạng trên cùng thuật toán định danh là NBNN [16].

Bảng 1: Số lượng đặc trưng trên các tập dữ liệu khác nhau

Tập dữ liệu	Số lượng phân lớp	Tổng số ảnh	Số đặc trưng trung bình trên mỗi ảnh	
			SIFT	DSIFT
AT&T	40	40	81	902
Face94	152	3040	153	1481
Face95	72	1440	94	853
Face96	151	3016	69	494

Từ Bảng 1, ta thấy số lượng đặc trưng DSIFT trung bình trên mỗi ảnh ở các tập dữ liệu khác nhau là lớn hơn số lượng đặc trưng SIFT rất nhiều (từ 7 lần trở lên).

Bảng 2: So sánh độ chính xác của thuật toán NBNN với SIFT và NBNN với DSIFT

Tập dữ liệu	Tỷ lệ nhận dạng (%)	
	NBNN (SIFT) [16] [4]	NBNN (DSIFT)
AT&T	99.37	99.67
Face94	100	100
Face95	98.83	99.63
Face96	99.20	99.80

Bảng 2 cho thấy, với số lượng lớn các đặc trưng DSIFT đã cải thiện đáng kể độ chính xác của thuật toán nhận dạng (độ chính xác tăng 0.3% trên tập ảnh AT&T, 0.8% trên Face95, 0.6% trên Face96). Tiếp theo, chúng tôi thực hiện đánh giá kết quả nhận dạng của phương pháp do chúng tôi

đề xuất trên các tập dữ liệu kiểm thử, được trình bày như trong Bảng 3.

Bảng 3: Tỷ lệ nhận dạng trên các tập dữ liệu

Tập dữ liệu	Số lượng phân lớp	Tổng số ảnh	Tỷ lệ nhận dạng (%)	
			NBNN	LNBNN (k=20)
AT&T	40	40	99.67	99.87
Face94	152	3040	100	100
Face95	72	1440	99.63	99.74
Face96	151	3016	99.80	99.84

Để việc đánh giá được thuyết phục hơn, chúng tôi so sánh kết quả của phương pháp đề xuất với các phương pháp tốt nhất mà chúng tôi biết hiện nay trong lĩnh vực nhận dạng mặt người, phương pháp BDPCA + LDA [8], LoG - DCT [12], DCT + HRBF [9], trên cùng tập dữ liệu. Kết quả nhận dạng khi thực nghiệm trên tập ảnh AT&T được trình bày trong Bảng 4.

Bảng 4: Tỷ lệ nhận dạng trên tập dữ liệu AT&T

Phương pháp	Tỷ lệ nhận dạng (%)
NBNN	99.67
LNBNN (k=20)	100
BDPCA + LDA	97.10
LoG - DCT	98.00
DCT + HRBF	97.68

Bảng kết quả nhận dạng trên tập dữ liệu AT&T cho thấy độ chính xác của NBNN và LNBNN khá cao, lần lượt là 99.67% và 100%. Kết quả tổng hợp trên các tập dữ liệu kiểm chuẩn, có so sánh với các phương pháp PCA + DCT [15], SVM based on LDA (Linear and RBF kernel) [23], PCA + LDA on Holistic Approach [13]; được trình bày như Bảng 5.

Kết quả từ Bảng 5 một lần nữa cho thấy thuật toán NBNN và LNBNN cho kết quả khá cao, trên 99%. Như vậy, từ kết quả thực nghiệm trên, ta có thể kết luận rằng thuật toán LNBNN cho kết quả tốt hơn các phương pháp khác trong hầu hết các tập dữ liệu (tuy nhiên, trên tập ảnh Face95 thuật toán LNBNN cho kết quả cao thứ hai, sau PCA + LDA on Holistic Approach). Bên cạnh đó, mô hình LNBNN là một cải tiến của NBNN nên không có quá trình học. Điều này giúp cho hệ thống nhận dạng đơn giản, dễ cài đặt nhưng lại cho kết quả với độ chính xác cao. Chúng tôi

Bảng 5: Tỷ lệ nhận dạng trên các tập dữ liệu kiểm chuẩn

Phương pháp	AT&T	Face94	Face95	Face96
NBNN	99.67	100	99.63	99.80
LNBNN	99.87	100	99.74	99.84
BDPCA+LDA	97.10	-	-	-
LoG - DCT	98.00	-	-	-
DCT + HRBF	97.68	-	-	-
PCA + DCT	91.30	99.90	87.00	94.00
SVM based on LDA (Linear)	-	90.00	-	90.00
SVM based on LDA (RBF kernel)	-	97.14	-	95.10
PCA + LDA on Holistic Approach	-	-	100	-

cũng đã đánh giá thời gian thực hiện của hệ thống với các phương pháp đề xuất qua các tập dữ liệu thực nghiệm. Kết quả được trình bày trong Bảng 6.

Bảng 6: Thời gian thực hiện nhận dạng

Tập dữ liệu	Thời gian nhận dạng (giây)	
	NBNN	LNBNN
AT&T	20.74	3.98
Face94	814.46	130.17
Face95	94.70	14.9
Face96	236.56	24.72

Các kết quả từ Bảng 6 cho thấy, thuật toán LNBNN không những cải thiện về độ chính xác nhận dạng mà còn rất hiệu quả về thời gian thực hiện (nhanh hơn từ 5 lần so với thuật toán NBNN).

VII. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Chúng tôi vừa trình bày sự kết hợp phương pháp phát hiện mặt người với Haar Like Features - Cascade of Boosted Classifiers, sự biểu diễn ảnh bằng các đặc trưng DSIFT, cùng với kỹ thuật định danh dùng thuật toán LNBNN. Kết quả thử nghiệm trên nhiều tập dữ liệu ảnh AT&T, Face94, Face95, Face96 cho thấy hệ thống do chúng tôi đề xuất cho phép nhận dạng khuôn mặt người một cách rất hiệu quả về độ chính xác cũng như thời gian thực hiện. Bên cạnh đó, việc rút trích khuôn mặt trước khi biểu diễn ảnh là cần thiết vì giúp loại bỏ các yếu tố nhiễu do ảnh nền (background

clutter) như nhà cửa, cây cối... Hơn nữa, việc biểu diễn ảnh bằng các đặc trưng DSIFT là rất cần thiết và hiệu quả: nếu chỉ xử lý ảnh thô thì chương trình sẽ hoàn toàn không nhận thấy được các điểm đặc biệt của khuôn mặt và do đó khả năng nhận dạng sai là rất cao. Điều này đã giúp bước định danh mặt người cải thiện đáng kể về tốc độ cũng như độ chính xác của thuật toán nhận dạng. Mặc dù cả hai thuật toán NBNN và LNBNN đều cho kết quả khả quan, chúng tôi phát hiện ra rằng LNBNN đã cải thiện đáng kể về mặt thời gian thực hiện và vì vậy có khả năng mở rộng cho các tập dữ liệu ảnh với số lượng lớn các lớp. Công việc kế tiếp của chúng tôi là thực hiện việc đánh giá thuật toán LNBNN trên tập dữ liệu “Labeled faces in the wild” với 5749 lớp và 13233 ảnh.

TÀI LIỆU THAM KHẢO

- [1] Lowe D G. Distinctive image features from Scale-Invariant keypoints. *International Journal of Computer Vision*. 2004;60(2):91–110. Available from: DOI: 10.1007/978-3-540-45243-0_39.
- [2] Bosch A, Zisserman A, Munoz X. Image classification using random forests and ferns. *In Proc ICCV*. 2007;Available from: DOI 10.1109/ICCV.2007.4409066.
- [3] McCann S, Lowe D G. Local Naive Bayes Nearest Neighbor for Image Classification. *Technical Report TR-2011-11*. 2011;University of British Columbia.
- [4] Kirby M, Sirovich L. A Low Dimensional Procedure for the Characterization of Human Faces. *Journal of the Optical Society of America A*. 1988;4(3):519–524.
- [5] Turk M A, Pentland A P. Face Recognition Using Eigenfaces. *IEEE*. 1991;.
- [6] Trần Phước Long, Nguyễn Văn Lượng. Nhận dạng người dựa vào thông tin khuôn mặt xuất hiện trên ảnh [Luận văn Cử nhân Tin học]. Trường Đại học Khoa học Tự nhiên; 2003. Tr. 39–41, 47–62.
- [7] Lu Boun Vinh, Hoàng Phương Anh. Nghiên cứu và xây dựng hệ thống nhận dạng mặt người dựa trên FSVM và AdaBoost [Luận văn Cử nhân Tin học]. Trường Đại học Khoa học Tự nhiên; 2011. Tr. 39–41, 47–62.
- [8] Zuo W, Zhang D, Yang J, Wang K. BDPCA plus LDA: a novel fast feature extraction technique for face recognition. *IEEE Transactions on Systems, Man, And Cybernetics – Part B: Cybernetics*. 2006;36(4).
- [9] Chen Y, Yaou Zhao. Face Recognition Using DCT and Hierarchical RBF Model. *School of Information Science and Engineering Jinan University*. 2006;Available from: DOI 10.1007/11875581_43.
- [10] Aly M. *Face Recognition Using SIFT Features*. Technical Report, Caltech; 2006.
- [11] Chennamma H R, Lalitha Rangarajan, Veerabhadrapa. Face Identification from Manipulated Facial Images using SIFT. *Department of Studies in Computer Science University of Mysore, Mysore, India*. 2011;Available from: Doi 10.1109/ICETET.2010.33.
- [12] Sharif M, Sajjad Mohsin, Muhammad Younas Javed, Muhammad Atif Ali. Single Image Face Recognition Using Laplacian of Gaussian and Discrete Cosine Transforms. *The International Arab Journal of Information Technology*. 2012;9(6).
- [13] Suhas S Satonkar, Kurhe Ajay B, Dr Prakash Khanale B. Face Recognition Using Principal Component Analysis and Linear Discriminant Analysis on Holistic Approach in Facial Images Database. *IOSR Journal of Engineering*. 2012;2(12):15–22.
- [14] Kumar H, Padmavati. Face Recognition using SIFT by varying Distance Calculation Matching Method. *International Journal of Computer Application*. 2012;47(3).
- [15] Sunil, Renke Pradnya. Automatic Face Recognition using Principal Component Analysis with DCT. *Journal of Electronics and Communication Engineering (IOSR-JECE)*. 2013;p. 01–07.
- [16] Châu Ngân Khánh, Đoàn Thanh Nghị. Nhận dạng mặt người với giải thuật Haar Like Feature – Cascade of Boosted Classifiers và đặc trưng SIFT. *Tạp chí Khoa học Trường Đại học An Giang*. 2014;3(2).
- [17] Ladislav Lenc, Pavel Král. Automatic face recognition system based on the SIFT features. *Computer and Electrical Engineering*. 2015;46.
- [18] Ehsan Sadeghipour, Nasrollah Sahragard. Face Recognition Based on Improved SIFT Algorithm. *International Journal of Advanced Computer Science and Applications*. 2016;7. Available from: DOI 10.1007/11875581_43.
- [19] Lienhart R, Kuranov A, Pisarevsky V. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. *MRL Technical Report*. 2002;Available from: DOI: 10.1007/978-3-540-45243-0_39.
- [20] Vedaldi A, Fulkerson B. *Dense SIFT as a faster SIFT*; 2017.
- [21] Oren Boiman, Eli Shechtman, Michal Irani. In Defense of Nearest-Neighbor Based Image Classification. *In CVPR*. 2008;.
- [22] Mitchell T M. *Machine Learning*. McGraw-Hill, New York, USA; 1997; tr. 128-153. Chapter 5.
- [23] Bouzalmat A, Jamal Kharroubi, Aرسالane Zarghili. Face Recognition Using SVM Based on LDA. *IJCSI International Journal of Computer Science Issues*. 2013;10(4).