**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN**

**QUÁ TRÌNH TÌNH TOÁN SỬ DỤNG EIGENFACES ĐỂ PHÂN TÍCH TRỰC TIẾP CÁC THÀNH PHẦN PCA**

Người hướng dẫn**: PGS. TS. NGUYỄN TẤN KHÔI**

Sinh viên thực hiện**:**

**Võ Hoàng Thái Bảo LỚP: 23T\_DT4**

**Dương Văn Chí Bảo LỚP: 23T\_DT4**

**Đà Nẵng, 2024**

MỤC LỤC

[MỞ ĐẦU i](#_heading=h.1fob9te)

[1.](#_heading=h.3znysh7) TỔNG QUAN ĐỀ TÀI 1

[2.](#_heading=h.2et92p0) CƠ SỞ LÝ THUYẾT 1

[2.1.](#_heading=h.tyjcwt) Ý tưởng 1

[2.2.](#_heading=h.3dy6vkm) Cơ sở lý thuyết 1

[3.](#_heading=h.1t3h5sf) TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 1

[3.1.](#_heading=h.4d34og8) Phát biểu bài toán 1

[3.2.](#_heading=h.2s8eyo1) Cấu trúc dữ liệu 1

[3.3.](#_heading=h.17dp8vu) Thuật toán 1

[4.](#_heading=h.3rdcrjn) CHƯƠNG TRÌNH VÀ KẾT QUẢ 1

[4.1.](#_heading=h.26in1rg) Tổ chức chương trình 1

[4.2.](#_heading=h.lnxbz9) Ngôn ngữ cài đặt 1

[4.3.](#_heading=h.35nkun2) Kết quả 1

[4.3.1.](#_heading=h.1ksv4uv) Giao diện chính của chương trình 1

[4.3.2.](#_heading=h.44sinio) Kết quả thực thi của chương trình 1

[4.3.3.](#_heading=h.2jxsxqh) Nhận xét đánh giá 1

[5.](#_heading=h.z337ya) KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 1

[5.1.](#_heading=h.3j2qqm3) Kết luận 1

[5.2.](#_heading=h.1y810tw) Hướng phát triển 1

[TÀI LIỆU THAM KHẢO 2](#_heading=h.4i7ojhp)

MỞ ĐẦU

Wwwwwwwwwww

Wwwwwwwww

Bắt đầu!!!!!!!!!

# TỔNG QUAN ĐỀ TÀI

+ PCA là phương pháp đơn giản nhất phân tích đa biến dựa trên các vector đặc trưng.

+ Phương pháp PCA cung cấp cho chúng ta một bức ảnh ít chiều, một cái bóng của vật thể khi quan sát từ chính những đặc trưng cơ bản nhất của vật thể đó.

+ Mục tiêu của phương pháp PCA là thực hiện giảm số chiều nhưng vẫn đảm bảo tối đa sự phân tán dữ liệu.

+ PCA là một thuật toán được sử dụng để tạo ra một ảnh mới từ ảnh ban đầu

+ Bản chất của PCA là tìm ra một không gian mới theo hướng biến thiên mạnh nhất của một tập hợp các vector trong không gian cho trước.

# CƠ SỞ LÝ THUYẾT

## Ý tưởng

Mục tiêu của phương pháp PCA là “giảm số chiều” của 1 tập vector sao cho vẫn đảm bảo được “tối đa thông tin quan trọng nhất” phương pháp PCA sẽ giữ lại K thuộc tính “mới” từ M các thuộc tính ban đầu (K<M).

## Cơ sở lý thuyết

**Một số khái niệm toán học**

**+ Độ lệch chuẩn**

Trong đó : n là số lượng phần tử trong tập hợp

là ký hiệu trung bình của mẫu

**+ Phương sai**

Phương sai chính là bình phương của độ lệch chuẩn

+ Hiệp phương sai + Ma trận hiệp phương sai

+ Ma trận đại số

+ Eigenvector (Vector riêng)

+ Eigenvalue (Giá trị riêng)

**Các bước cơ bản trong PCA:**

Bước 1: Lấy dữ liệu (Get data)

Bước 2: Trừ trung bình mẫu. Với mỗi chiều dữ liệu giả sử ở chiều x, ta đều có 1 trung bình mẫu, công việc trong bước này là trừ tất cả giá trị trong chiều x cho trung bình mẫu x. Kết thúc bước này ta sẽ có trung bình mẫu ở tất cả các chiều là 0.

Bước 3: Tính ma trận hiệp phương sai

Bước 4: Tính các vectơ riêng và giá trị riêng của ma trận hiệp phương sai.

Bước 5: Chọn các thành phần chính

Đây là bước cuối cùng trong PCA. Trong bước này, tùy thuộc vào số lượng thành phần chính cần lấy, ta lấy lần lượt các thành phần (vectơ riêng) tương ứng với các giá trị riêng cao nhất.

**Eigenfaces trong nhận dạng cảm xúc trên khuôn mặt**

**Giới thiệu**

+ Eigenfaces là một phương pháp đã được nghiên cứu về nhận dạng trên khuôn mặt dựa trên phân tích các thành phần chính (PCA).

+ Eigenfaces là phương pháp áp dụng trực tiếp phép phân tích các thành phần PCA, nó đã được áp dụng rất nhiều vào biểu diễn, phát hiện và nhận dạng khuôn mặt.

Các bước cơ bản trong Eigenfaces:

+ Xử lý đầu vào:

Chuyển ảnh màu sang ảnh xám

Tạo cơ sở dữ liệu dựa trên các đặc trưng trên khuôn mặt như mắt, mũi và miệng.

**+ Huấn luyện ảnh:**

Tính vector khuôn mặt trung bình theo công thức

Trừ vector ảnh đầu vào cho vector ảnh khuôn mặt trung bình được

**Tính ma trận phương sai (Covariance) C:**

A sẽ có kích thước N2 x M

Tính các vector riêng u

Chỉ giữ lại K vector riêng trong số M vector nói trên với K được xác định như sau:

. Sắp xếp theo thứ tự dãy giảm dần các eigenvalues tìm được.

Theo dõi sự biến thiên của dãy trên, khi không còn biến thiên (hoặc xấp xỉ bằng không) thì lúc đó ta đã chọn đủ K.

Biểu diễn các khuôn mặt có sẵn.

# TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

## Phát biểu bài toán

* Input: Một ma trận 4X9
* Outout: Ma trận rút gọn của ma trận Input bằng PCA

## Cấu trúc dữ liệu

## Thuật toán

Trình bày các thuật toán và phân tích độ phức tạp của các thuật toán.

# CHƯƠNG TRÌNH VÀ KẾT QUẢ

## Tổ chức chương trình

#include <bits/stdc++.h>

using namespace std;

const long MAX\_ARRAY = 1e3;

//Buoc 3: Tinh su chenh lech giua bien ban dau va gia tri trung binh tim duoc o buoc 2

vector <vector<float>> step3(int N, int M, vector <vector<float>> T, vector <float> a)

{

vector <vector<float>> data;

data.resize(N, vector<float>(M)); //N dong M cot

for (int i = 0; i < N; i++)

for (int j = 0; j < M; j++)

data[i][j] = T[i][j] - a[i];

return data;

}

//Buoc 2: Tinh vector trung binh

vector <float> step2(int N, int M, vector <vector<float>> T)

{

vector <float> average;

for (int i = 0; i < N; i++)

{

float sum = 0;

for (int j = 0; j < M; j++)

sum += T[i][j];

average.push\_back(sum / (float) M);

}

return average;

}

// Buoc 1: Doc du lieu

int main()

{

//Khai bao

int N, M;

vector <vector<float>> T, tempData;

vector <float> a;

//Doc du lieu tu file data.inp vao, xuat ra file data.out

freopen("data.inp", "r", stdin);

freopen("data.out", "w", stdout);

//Doc du lieu dau vao

cin >> M >> N;

//Cap phat bo nho

T.resize(N, vector<float>(M)); //N dong M cot

tempData.resize(N, vector<float>(M)); //N dong M cot

//Doc gia tri cho mang T

for (int i = 0; i < N; i++)

for (int j = 0; j < M; j++)

cin >> T[i][j];

a = step2(N, M, T);

tempData = step3(N, M, T, a);

//In thu du lieu

// for (int i = 0; i < N; i++)

// {

// for (int j = 0; j < M; j++)

// cout << tempData[i][j] << " ";

// cout << endl;

// }

}

## Ngôn ngữ cài đặt

## Kết quả

### Giao diện chính của chương trình

### Kết quả thực thi của chương trình

Mô tả kết quả thực hiện chương trình.

### Nhận xét đánh giá

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

## Hướng phát triển

TÀI LIỆU THAM KHẢO

<https://www.academia.edu/34931265/Eigenfaces_-_LE_HOANG_THANH_1_> [https://www.turing.com/kb/guide-to-principal-component-analysis](https://www.turing.com/kb/guide-to-principal-component-analysis%20) [https://www.gatevidyalay.com/tag/principal-component-analysis-numerical-example/](https://www.gatevidyalay.com/tag/principal-component-analysis-numerical-example/%20) [https://github.com/toosyou/PCA/tree/master](https://github.com/toosyou/PCA/tree/master%20) <https://machinelearningcoban.com/2017/06/15/pca/>

PHỤ LỤC

Sinh viên bỏ Code từng phần vào đây.{*Font: Time New Roman; thường; cỡ chữ: 12; dãn dòng: 1,3; căn lề: justified}*