**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

----------------------------------------

**BÁO CÁO   
THỰC TẬP TỐT NGHIỆP**

***Đề tài*: “Nghiên cứu hệ thống thư viện OpenCV và xây dựng ứng dụng nhận diện thông qua Xử lý ảnh”.**

**Người hướng dẫn : TS.GVC: VÕ XUÂN THỂ.**

**Sinh viên thực hiện : NGUYỄN XUÂN HIẾU .**

**Mã số sinh viên : N14DCCN124.**

**Lớp : D14CQCP01**

**Khoá** **: 2014**

**Hệ** **: ĐẠI HỌC CHÍNH QUY.**

**TP.HCM, tháng 7 / 2018 .**

NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP

............................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................ .................................................................................................................................  
.... ...................................................................................................................................................  
............................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................. ...................................................................................................................................................... ................................................................................................................................................................................................................................................................................................................... .................................................................................................................................................. ..................................................................................................................................................................................... ......................................................................................................................... ............................................................................................................................................................. ................................................................................................................................................ ............................................................................................................................................................................................................................................................................................................................................................ ......................................................................................................... ................................................................................................................................................................................................................................................................

Tp HCM, ngày .../7/2018.

LỜI NÓI ĐẦU

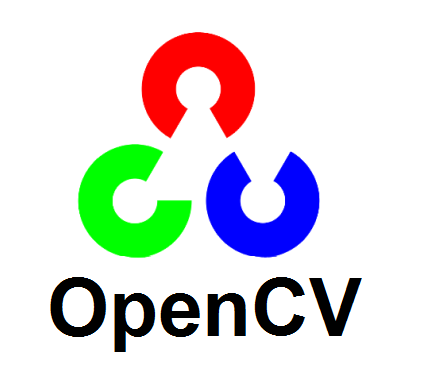
Xử lý ảnh và thị giác máy là lĩnh vực mà ngày nay được phát triển và ứng dụng rất rộng rãi trong nhiều lĩnh vực khác nhau nhờ vào sự phát triển ngày càng mạnh mẽ của các hệ thống máy tính, các thuật toán và công trình nghiên cứu khác nhau của nhiều nhà khoa học trên thế giới.

Ở Việt Nam, các ứng dụng về xử ảnh đã bước đầu được triển khai trên một số lĩnh vực như lắp đặt hệ thống nhận dạng biển biển số xe ở các bãi đổ xe, hệ thống nhận dạng vân tay chấm công ở các công sở ... Thông qua OpenCV (Open Source Computer Vision) , một thư viện mã nguồn mở về thị giác máy với các thuật toán đã được tối ưu về xử lý ảnh và các vấn đề liên quan tới thị giác máy, chúng ta có thể sáng tạo và ứng dụng linh hoạt các thuật toán cơ bản lên ảnh và Video số cung cấp cho người dùng những công cụ tiện dụng trong việc thao tác và sử dụng ảnh như: lọc ảnh, chỉnh sửa ảnh, ... Đặc biệt là áp dụng trong kỹ thuật nhận diện ảnh trong thực tế hiện nay, vậy ta   
hãy cùng nhau tìm hiểu sơ lược qua vấn đề này.

**CHƯƠNG I: GIỚI THIỆU VỀ THƯ VIỆN OPENCV.**

**1. Làm quen với OpenCV:**

**-** OpenCV (Open Source Computer Vision) là một thư viện mã nguồn mở về thị giác máy với hơn 500 hàm và hơn 2500 các thuật toán đã được tối ưu về xử lý ảnh, và các vấn đề liên quan tới thị giác máy. OpenCV được thiết kế một cách tối ưu, sử dụng tối đa sức mạnh của các dòng chip đa lõi… để thực hiện các phép tính toán trong thời gian thực, nghĩa là tốc độ đáp ứng của nó có thể đủ nhanh cho các ứng dụng thông thường.

 ***H.1.1***

- Dự án về OpenCV được khởi động từ những năm 1999, đến năm 2000 nó được giới thiệu trong một hội nghị của IEEE về các vấn đề trong thị giác máy và nhận dạng, tuy nhiên bản OpenCV 1.0 mãi tới tận năm 2006 mới chính thức được công bố và năm 2008 bản 1.1 (pre-release) mới được ra đời. Tháng 10 năm 2009, bản OpenCV thế hệ thứ hai ra đời (thường gọi là phiên bản 2.x), phiên bản này có giao diện của C++ (khác với phiên bản trước có giao diện của C) và có khá nhiều điểm khác biệt so với phiện bản thứ nhất.

- Được chấp nhận trên toàn thế giới, OpenCV có hơn 47 nghìn người dùng cộng đồng người dùng và số lượt tải xuống ước tính vượt quá 14 triệu lượt tải xuống.

**2. Nền tảng cài đặt.**

- OpenCV được thiết kế để chạy trên nhiều nền tảng khác nhau (cross-patform), nghĩa là nó có thể chạy trên hệ điều hành Window, Linux, Mac, iOS … Việc sử dụng thư viện OpenCV tuân theo các quy định về sử dụng phần mềm mã nguồn mở BSD do đó chúng ta có thể sử dụng thư viện này một cách miễn phí cho cả mục đích phi thương mại lẫn thương mại.

**3. Các ứng dụng thực tế.**

- Opencv có rất nhiều ứng dụng như :

+ Nhận dạng ảnh.

+ Xử lý hình ảnh( lọc khử nhiễu, chỉnh sửa, phân tích và nén ảnh, video,...)

+ Phục hồi hình ảnh/video.

+ Chuẩn đoán ảnh, 3D, Thực tế ảo.

+ Bảo mật (qua xử lý dấu vân tay hay mống mắt).

+ Xe tự lái và các ứng dụng khác:

- Cùng với các công ty có uy tín như Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota sử dụng thư viện, có nhiều công ty khởi nghiệp như Applied Minds, VideoSurf và Zeitera, sử dụng rộng rãi OpenCV. Các ứng dụng được triển khai của OpenCV mở rộng phạm vi từ khâu hình ảnh đường phố, phát hiện xâm nhập trong video giám sát ở Israel, giám sát thiết bị mỏ ở Trung Quốc, giúp robot điều hướng và  
nhận đồ vật tại Willow Garage, phát hiện tai nạn ở châu Âu. Tây Ban Nha và New York, kiểm tra đường băng cho các mảnh vỡ ở Thổ Nhĩ Kỳ, kiểm tra nhãn trên các sản phẩm tại các nhà máy trên khắp thế giới để phát hiện khuôn mặt nhanh chóng ở Nhật Bản.

****



***H.1.2 H.1.3***

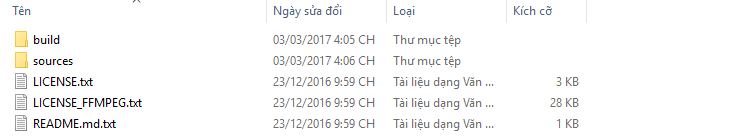
**4. Cài đặt .**

**a)** trên môi trường Windows, ngôn ngữ sử dụng C/C++.

**1**. Trước tiên, tải gói cài đặt miễn phí tại :  
<https://sourceforge.net/projects/opencvlibrary/>

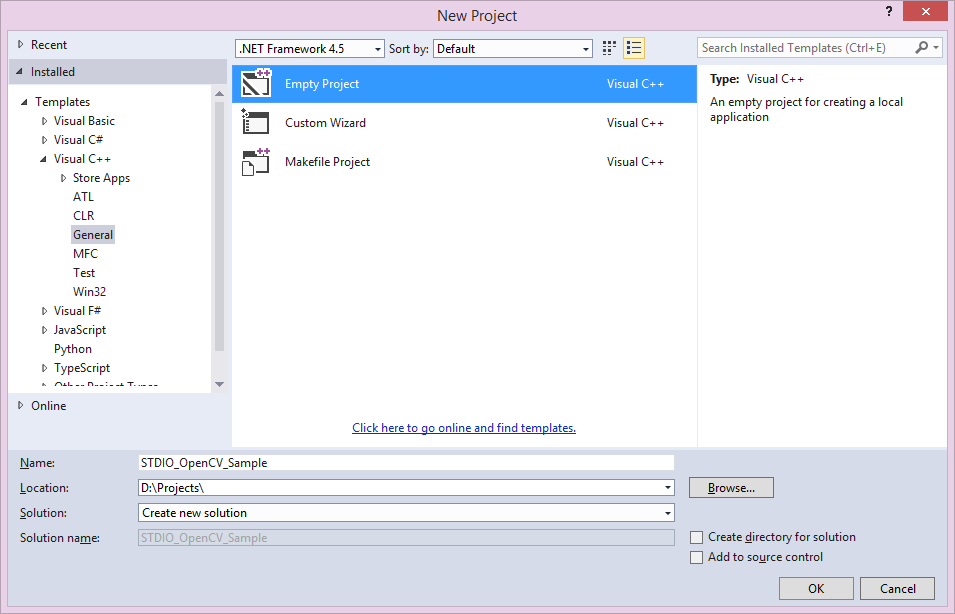
**2**. Thư viện OpenCV phiên bản 3.4.2 sau khi tải về có dạng **opencv-3.4.2.exe**. Tiến hành chạy file cài đặt, ta chọn đường dẫn để bộ cài đặt giải nén thư viện OpenCV.

- Sau khi giải nén, ta thu được 2 folder:

* **Build**: chứa thư viện OpenCV đã được biên dịch thành thư viện cho nhiều ngôn ngữ khác nhau: C++, Java, Python.
* **Sources**: tài liệu hướng dẫn, ví dụ và mã nguồn của OpenCV.

***H.1.4***

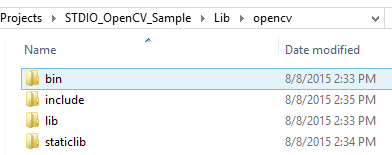
**3**. Tích hợp vào Project:

- Khởi tạo 1 project trống:   


***H.1.5***

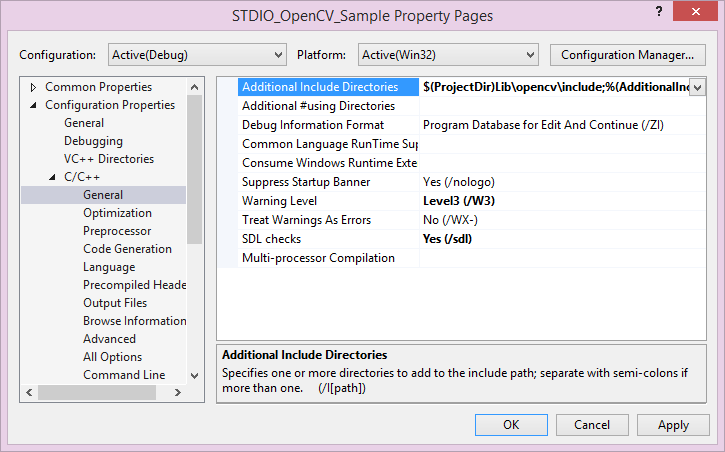
- Tại thư mục chứa project, ta tạo folder **Lib/opencv** và copy toàn bộ nội dung ở đường dẫn **<extracted\_path>/build/x86/vc12** và **<extracted\_path>/build/include**   
vào. Trong đó, **<extracted\_path>** là đường dẫn cài đặt OpenCV ở trên.

- Sau khi copy, ta có được cấu trúc folder như sau:



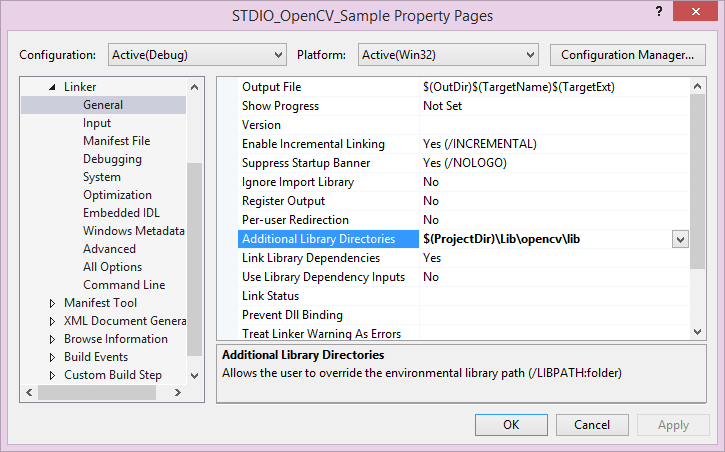
***H.1.6***

- Tại giao diện của Visual Studio, chọn **Project Properties > Configuration Properties > C/C++ > General > Addition Include Directories**, ta thêm đường dẫn **$(ProjectDir)\Lib\opencv\include** .



H.1.7

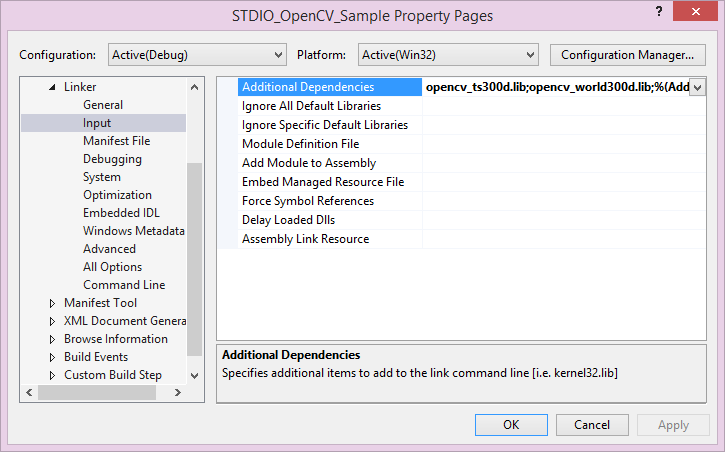
- Tiếp theo, cũng tại trang Project Properties, chọn **Linker > General > Addition Library Directories**, thêm vào đường dẫn **$(ProjectDir)\Lib\opencv\lib** .



***H.1.8***

- Và tại **Linker > Input**, ta thêm vào các thư viện sau đây:



H.1.9

- Cuối cùng, ta copy file **opencv\_world300d.dll** trong **Lib\opencv\bin** vào folder **Debug**của project. Tới đây, ta đã kết thúc việc tích hợp thư viện OpenCV vào project. Để kiểm tra tính chính xác của công việc ta đã làm, ta sẽ viết một ví dụ nhỏ sử dụng OpenCV.

**4**. Minh họa sử dụng OpenCV:

- Trong ví dụ minh họa này, chương trình của ta có khả năng:

* Load một ảnh bất kì
* Đọc vào đối tượng Mat của OpenCV
* Hiển thị lên màn hình

Ta tạo file **sample.cpp** có nội dung như sau:

1. #include <stdio.h>
2. #include "opencv2/core/core.hpp"
3. #include "opencv2/highgui/highgui.hpp"
5. using namespace cv;
7. int main()
8. {
9. // Read image
10. Mat image = imread("Lena.png", CV\_LOAD\_IMAGE\_COLOR);
12. // Check for valid
13. if (!image.data)
14. {
15. printf("Could not open or find the image\n");
16. return -1;
17. }
19. // Create and show image in window
20. imshow(" OpenCV Sample", image);
22. // Wait input and exit
23. waitKey(0);
25. return 0;
26. }

**Dòng 10:**đọc file ảnh lên với hàm imread cùng tùy chọn CV\_LOAD\_IMAGE\_ COLOR (đọc ảnh màu).   
  
 - Hàm này trả về một ma trận các pixel màu. Ý nghĩa của các tùy chọn

* CV\_LOAD\_IMAGE\_UNCHANGED: load ảnh bình thường (bao gồm cả kênh alpha nếu có)
* CV\_LOAD\_IMAGE\_GRAYSCALE: load ảnh và chuyển về trắng đen
* CV\_LOAD\_IMAGE\_COLOR: load ảnh phần màu (RGB), bỏ qua kênh alpha

**Dòng 13 – 17:** kiểm tra việc đọc ảnh có thành công hay không.

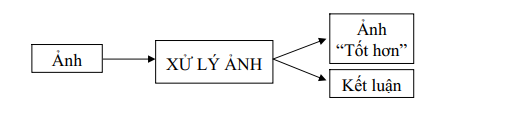
**Dòng 20:**  Hiển thị ảnh đã load sử dụng hàm imshow với tham số là tiêu đề cửa sổ và nội dung ảnh.

**CHƯƠNG II : CÁC KHÁI NIỆM VỀ XỬ LÝ ẢNH SỐ.**

**1. Xử lý ảnh là gì.**

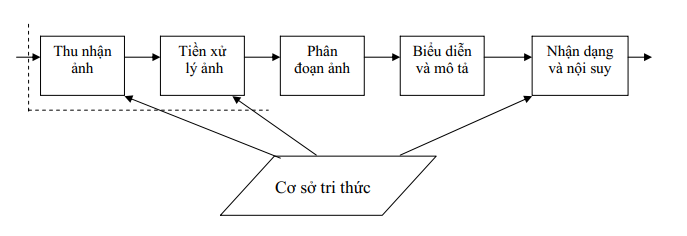
- Con người thu nhận thông tin qua các giác quan, trong đó thị giác  
đóng vai trò quan trọng nhất. Những năm trở lại đây với sự phát triển của  
phần cứng máy tính, xử lý ảnh và đồ hoạ đó phát triển một cách mạnh mẽ  
và có nhiều ứng dụng trong cuộc sống

- Quá trình xử lý ảnh được xem như là quá trình thao tác ảnh đầu vào  
nhằm cho ra kết quả mong muốn. Kết quả đầu ra của một quá trình xử lý  
ảnh có thể là một ảnh “tốt hơn” hoặc một kết luận.



**H.2.1**

**2. Các bước cơ bản trong xử lý ảnh.**

** H.2.2**

**a) Phần thu nhận ảnh:  
  
 -** Ảnh có thể nhận qua camera màu hoặc đen trắng. Thường ảnh nhận qua camera là ảnh tương tự (loại camera ống chuẩn CCIR với tần số 1/25, mỗi ảnh 25 dòng), cũng có loại camera đã số hoá (như loại CCD – Change Coupled Device) là loại photodiot tạo cường độ sáng tại mỗi điểm ảnh.

**b) Tiền xử lý ảnh:** - Sau bộ thu nhận, ảnh có thể nhiễu độ tương phản thấp nên cần đưa vào bộ tiền xử lý để nâng cao chất lượng. Chức năng chính của bộ tiền xử lý là lọc nhiễu, nâng độ tương phản để làm ảnh rõ hơn, nét hơn.

**c) Phân đoạn (Segmentation) hay phân vùng ảnh:** - Phân vùng ảnh là tách một ảnh đầu vào thành các vùng thành phần để biểu diễn phân tích, nhận dạng ảnh. Ví dụ: để nhận dạng chữ (hoặc mã vạch) trên phong bì thư cho mục đích phân loại bưu phẩm, cần chia các câu, chữ về địa chỉ hoặc tên người thành các từ, các chữ, các số (hoặc các vạch) riêng biệt để nhận dạng. Đây là phần phức tạp khó khăn nhất trong xử lý ảnh và cũng dễ gây lỗi, làm mất độ chính xác của ảnh. Kết quả nhận dạng ảnh phụ thuộc rất nhiều vào công đoạn này.

**d) Biểu diễn ảnh (Image Representation) :** - Đầu ra ảnh sau phân đoạn chứa các điểm ảnh của vùng ảnh (ảnh đã phân đoạn) cộng với mã liên kết với các vùng lận cận. Việc biến đổi các số liệu này thành dạng thích hợp là cần thiết cho xử lý tiếp theo bằng máy tính. Việc chọn các tính chất để thể hiện ảnh gọi là trích chọn đặc trưng (*Feature Selection*) gắn với việc tách các đặc tính của ảnh dưới dạng các thông tin định lượng hoặc làm cơ sở để phân biệt lớp đối tượng này với đối tượng khác trong phạm vi ảnh nhận được. Ví dụ: trong nhận dạng ký tự trên phong bì thư, chúng ta miêu tả các đặc trưng của từng ký tự giúp phân biệt ký tự này với ký tự khác.

**e) Nhận dạng và nội suy ảnh (Image Recognition and Interpretation):** - Nhận dạng ảnh là quá trình xác định ảnh. Quá trình này thường thu được bằng cách so sánh với mẫu chuẩn đã được học (hoặc lưu) từ trước. Nội suy là phán đoán theo ý nghĩa trên cơ sở nhận dạng. Ví dụ: một loạt chữ số và nét gạch ngang trên phong bì thư có thể được nội suy thành mã điện thoại. Có nhiều cách phân loai ảnh khác nhau về ảnh. Theo lý thuyết về nhận dạng, các mô hình toán học về ảnh được phân theo hai loại nhận dạng ảnh cơ bản:  
 + Nhận dạng theo tham số.  
 + Nhận dạng theo cấu trúc.  
Một số đối tượng nhận dạng khá phổ biến hiện nay đang được áp dụng trong khoa học và công nghệ là: nhận dạng ký tự (chữ in, chữ viết tay, chữ ký điện tử), nhận dạng văn bản (Text), nhận dạng vân tay, nhận dạng mã vạch, nhận dạng mặt người…

**3. Những vấn đề cơ bản trong hệ thống xử lý ảnh.**

***3.1 Điểm ảnh (Picture Element):***

a) *Định nghĩa:* Điểm ảnh (Pixel) là một phần tử của ảnh số tại toạ độ *(x, y)* với độ xám hoặc màu nhất định. Kích thước và khoảng cách giữa các điểm ảnh đó được chọn thích hợp sao cho mắt người cảm nhận sự liên tục về không gian và mức xám (hoặc màu) của ảnh số gần như ảnh thật. Mỗi phần tử trong ma trận được gọi là một phần tử ảnh.

***3.2 Độ phân giải của ảnh.***

a) *Định nghĩa*: Độ phân giải (Resolution) của ảnh là mật độ điểm ảnh được ấn định trên một ảnh số được hiển thị.

- Ví dụ: 320x240 tức là ảnh trên màn hình chiều dọc có 320 điểm ảnh, còn chiều   
ngang có 240 điểm ảnh.

=> Nhận xét : cùng một mật độ (độ phân giải) nhưng diện tích màn hình rộng hơn thì độ mịn (liên tục của các điểm) kém hơn.

***3.3******Mức xám của ảnh.***

- Một điểm ảnh (pixel) có hai đặc trưng cơ bản là vị trí *(x, y)* của điểm ảnh và độ xám của nó. Dưới đây chúng ta xem xét một số khái niệm và thuật ngữ thường dùng trong xử lý ảnh:

*a) Định nghĩa*: Mức xám của điểm ảnh là cường độ sáng của nó được gán bằng giá trị số tại điểm đó.  
  
 *b) Các thang giá trị mức xám thông thường:*16, 32, 64, 128, 256 (Mức 256 là mức phổ dụng. Lý do: từ kỹ thuật máy tính dùng 1 byte (8 bit) để biểu diễn mức xám: Mức xám dùng 1 byte biểu diễn: 28=256 mức, tức là từ 0 đến 255).  
  
*c) Ảnh đen trắng:*là ảnh có hai màu đen, trắng (không chứa màu khác) với mức xám ở các điểm ảnh có thể khác nhau.  
  
*d) Ảnh nhị phân:*ảnh chỉ có 2 mức đen trắng phân biệt tức dùng 1 bit mô tả 2 mức khác nhau. Nói cách khác: mỗi điểm ảnh của ảnh nhị phân chỉ có thể là 0 hoặc 1.  
  
 *e) Ảnh màu:*trong khuôn khổ lý thuyết ba màu (Red, Blue, Green) để tạo nên thế giới màu, người ta thường dùng 3 byte để mô tả mức màu, khi đó các giá trị màu: 28\*3=224 ≈ 16,7 triệu màu.

**3.4 Định nghĩa ảnh số.**

**\*** Ảnh số là tập hợp các điểm ảnh với mức xám phù hợp dùng để mô tả ảnh gần với ảnh thật.

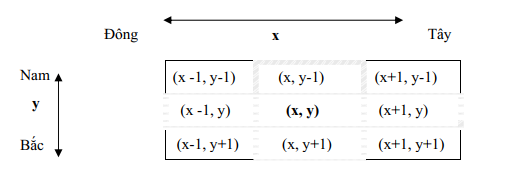
**3.5 Quan hệ giữa các điểm ảnh.**

*a) Các lân cận của điểm ảnh (Image Neighbors):*

- Giả sử có điểm ảnh p tại toạ độ (x, y). p có 4 điểm lân cận gần nhất theo chiều đứng và ngang (có thể coi như lân cận 4 hướng chính: Đông, Tây, Nam, Bắc).

+ {(x-1, y); (x, y-1); (x, y+1); (x+1, y)} = N4(p) .

(trong đó: số 1 là giá trị logic; N4(p) tập 4 điểm lân cận của p).



**H 2.3**

*b) khoảng cách giữa các điểm ảnh:*

- Khoảng cách Euclide : Khoảng cách Euclide giữa hai điểm ảnh *p(x, y)* và *q(s, t)* được định nghĩa như sau:

*De(p, q) = [(x - s)2 + (y - t)2]1/2 .*

**CHƯƠNG III : NHẬN DẠNG ẢNH.**

**1 . Khái niệm nhận dạng.**

**1.1 Khái niệm:**

- Nhận dạng là quá trình phân loại các đối tượng được biểu diễn theo một mô hình nào đó và gán chúng một tên (gán cho đối tượng một tên gọi, tức là một dạng) dựa theo những quy luật và mẫu chuẩn. Quá trình nhận dạng dựa vào những mẫu học biết trước gọi là *nhận dạng có thầy* hay học có thầy, trong những trường hợp ngược lại gọi *là học không có thầy*.   
  
 - *Nhận dạng ảnh* là giai đoạn cuối của các hệ thống xử lý ảnh. Nhận dạng ảnh dựa trên lý thuyết nhận dạng (Pattern Recognition) đã được đề cập trong nhiều sách về nhận dạng. Trong lý thuyết về nhận dạng nói chung và nhận dạng ảnh nói riêng có ba cách tiếp cận khác nhau:   
  
 a) Nhận dạng dựa vào phân hoạch không gian.  
  
 b) Nhận dạng dựa vào cấu trúc.  
  
 c) Nhận dạng dựa vào kỹ thuật mạng nơron.  
  
 => Hai cách tiếp cận đầu là cách tiếp cận kinh điển. Các đối tượng ảnh quan sát và thu nhận được phải trải qua giai đoạn tiền xử lý nhằm tăng cường chất lượng, làm nổi các chi tiết, tiếp theo là trích chọn và biểu diễn các đặc trưng, cuối cùng mới là giai đoạn nhận dạng. Cách tiếp cận thứ ba hoàn toàn khác. Nó dựa vào cơ chế đoán nhận, lưu trữ và phân biệt đối tượng mô phỏng theo hoạt động của hệ thần kinh con người. Do cơ chế đặc biệt, các đối tượng thu nhận bởi thị giác người không cần qua giai đoạn cải thiện mà chuyển ngay sang giai đoạn tổng hợp, đối sánh với các mẫu đã lưu trữ để nhận dạng.

- Trong phần này do điều kiện và thời gian chưa đủ nên phần báo cáo chỉ tập trung nghiên cứu phương pháp thứ nhất, cụ thể là phương pháp nhận dạng mặt người ***PCA***   
(Principal component analysis – phân tích thành phần chính).

**CHƯƠNG IV : MÔ PHỎNG CHƯƠNG TRÌNH**

**I. Công cụ sử dụng và một số file liên quan**

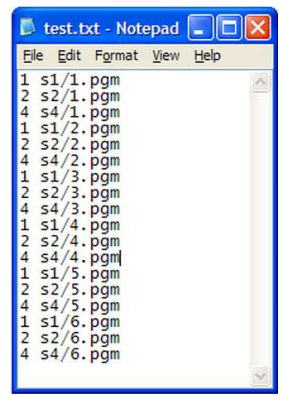
**1.1. Thư viện Opencv 2.4.3**

- Là thư viện hỗ trợ xử lý hình ảnh

**1.2. Công cụ Visual studio C++ 2010 Express.**  
 - Chúng ta sẽ sử dụng công cụ này để viết ứng dụng bằng C . - Công cụ này cần được cộng thêm thư viện của opencv 2.4.3

**1.3. Bộ dữ liệu sử dụng**

- Hai file : + train.txt : lưu thông tin những ảnh được đưa ra để huấn luyện (bao gồm chỉ số ảnh và đường dẫn đến ảnh)

****

**II. Các biến , hàm trong sử dụng trong chương trình**

**2.1. Chức năng tìm khuôn mặt giống với khuôn mặt cho trước Sử dụng các biến toàn cục :**

- int nTrainFaces : số ảnh được đưa ra để huấn luyện

- int nEigens : số các giá trị riêng

- IplImage \*\*faceImgArr : mảng chứa các ảnh

- IplImage \*pAvgTrainImg : ảnh trung bình

- IplImage \*\*eigenVecArr : vector riêng

- CvMat \*personNumTruthMat : mảng chứa các chỉ số ảnh

- Cv \*projectedTrainFaceMat : lưu trữ các khuôn mặt huấn luyện sau khi chiếu lên không gian con PCA Sử dụng các hàm :

- learn() : thực hiện công việc

+ load dữ liệu (chỉ số và đường dẫn ảnh để huấn luyện trong file train.txt) + thực hiện thuật toán PCA để tìm không gian con .

+ chiếu dữ liệu huấn luyện lên không gian .

+ lưu lại tất cả các thông tin (giá trị riêng, vector riêng, ảnh trung bình, chỉ số ảnh , khuôn mặt được chiếu).

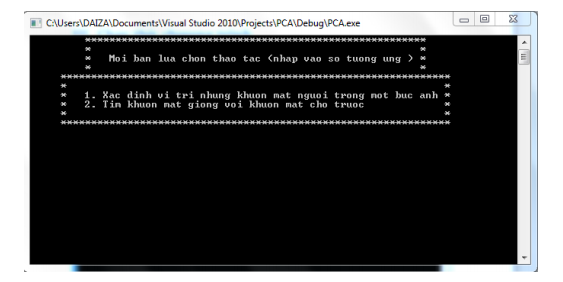
2.2. Chức năng tìm và đánh dấu khuôn mặt có trong ảnh .

Sử dụng các biến toàn cục:

- CvHaarClassifierCascade \*pCascade : load dữ liệu từ file XML trong opencv - CvMemStorage \*pStorage : Bộ nhớ đệm tự động.

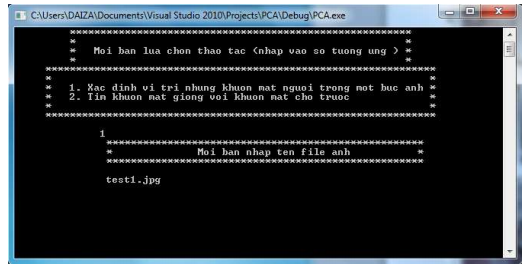
**III. Chạy thử chương trình**

Sau khi chạy chương trình , màn hình xuất hiện.



**H.4.3**

- > Ta chọn chức năng 1, màn hình yêu cầu nhập tên file ảnh :



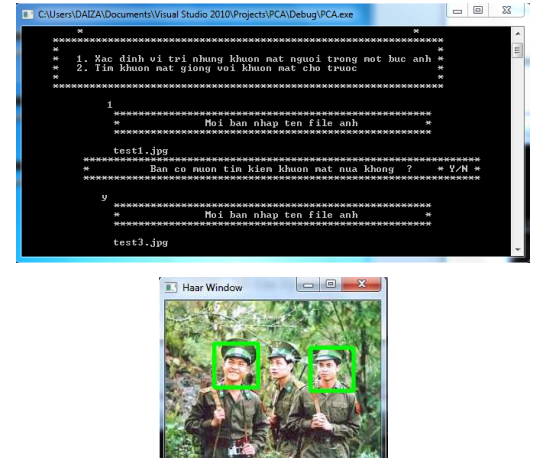
**H.4.4**

Nhập tên file ảnh , chương trình cho kết quả



**H.4.5**

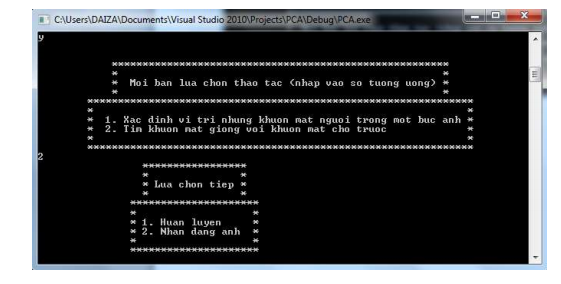
Tắt cửa sổ ảnh , chương trình hỏi bạn có muốn tiếp tục tìm kiếm khuôn mặt nữa không, bạn chọn “y” và nhập tên file ảnh và lặp lại quá trình trên .



**H.4.6**

- Nếu không muốn thực hiện thao tác này ta chọn “n” , chương trình hỏi bạn có muốn tiếp tục với chức năng khác không , nếu chon “n” chương trình sẽ dừng , chọn “y” chương trình trở về menu ban đầu .

- Ta chọn chức năng 2:



**H.4.7**

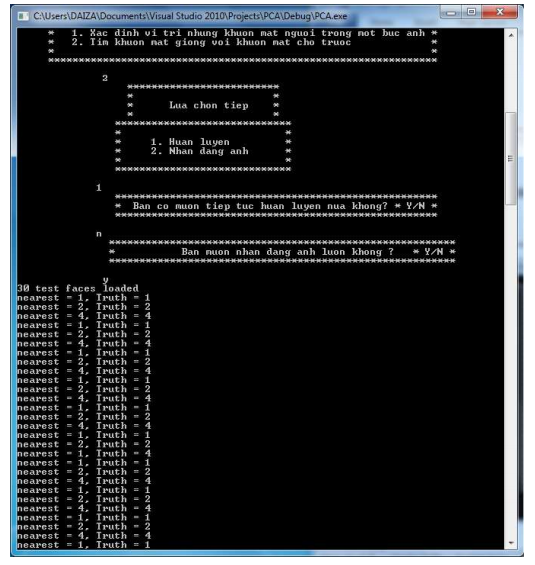
- Chọn tiếp chức năng 1. Chương trình hỏi tiếp ta nhập “n”



**H.4.8**

- sau đó nhập “y” để đồng ý nhận dạng ảnh, chương trình hiển thị kết quả .

Chức năng 2 trong Mục lựa chọn tiếp cũng cho kết quả tương tự .

****

**CHƯƠNG V: KẾT LUẬN**

**I. Đánh giá thuật toán.**

PCA là phương pháp dùng để xây dựng đặc trưng khuôn mặt , nó là một nền tảng đẻ phát triển các thuật toán khác. Thực tế có rất nhiều tác giả đã áp dụng thuật toán PCA, kết hợp với những phương pháp như mạng neuralb , xác xuất thông kê để xây dựng những thuật toán có độ chính xác cao .

Ưu điểm của thuật toán PCA là nó dễ cài đặt , nếu ta xét bài toàn tìm khuôn mặt giống nhau thì chỉ cần áp dụng hoàn toàn vào lý thuyết là đã có độ chính xác khá cao, nếu áp dụng cho bài toán tìm vị trí khuôn mặt thì cần phải có một thuật toán nữa để sử dụng face map thu được theo lý thuyết . Khuyết điểm của PCA là các mẫu khuôn mặt luân phụ thuộc vào tập huấn luyện , có nghĩa là các khuôn mặt trong ảnh kiểm tra phải giống với các ảnh huấn luyện về kích thước , tư thế , độ sáng . Thực tế trong tập huấn luyện gồm nhiều nhóm hình , mỗi nhóm hình là hình của một người với tư thế và độ sáng khác nhau . Có rất nhiều thuật toán nhận dạng mặt người dựa trên những chi tiết của khuôn mặt như đôi mắt , lông mày, cấu trúc của khuôn mặt, hay màu da.

**II. So sánh với các thuật toán khác.**

- Cách nhận diện khuôn mặt sử dụng phương pháp PCA phụ thuộc rất nhiều vào cơ sở dữ liệu ban đầu chứa các ảnh mẫu và góc quay camera cũng như ánh sáng. Sử dụng các thuật toán đại số để tìm giá trị mặt riêng và vector riêng rồi so sánh với giá trị mẫu, ta thu được khuôn mặt cần nhận diện.

- Đặc điểm của phương pháp này là giảm thiểu được dữ liệu cần sử dụng làm mẫu. Trong khi đó, phương pháp LDA lại phân loại các lớp chưa biết thành các lớp đã biết, mà ở đó các khuôn mặt tạo thành một lớp và sự khác biệt giữa các khuôn mặt trong một lớp là rất nhỏ. Cả PCA và LDA đều chọn cách thống kê lấy mẫu, chọn lọc để nhận diện khuôn mặt.

- Phương pháp còn lại EBGM chia mặt thành mạng lưới gồm các nút với mỗi khuôn mặt có khoảng 80 điểm nút. Vị trí của các nút giúp xác định khoảng cách giữa hai mắt, độ dài của sống mũi, độ sâu của hốc mắt, hình dạng của gò má… Điểm khó của phương pháp này là cần tính toán chính xác khoảng cách giữa các điểm nút, và do đó đôi khi nó phải dùng kết hợp với các phương pháp như PCA hay LDA. Với những hạn chế khi sử dụng công nghệ nhận diện khuôn mặt truyền thống, phương pháp nhận diện 3D đã trở thành hướng đi mới trong việc ứng dụng công nghệ nhận diện khuôn mặt.

**TÀI LIÊU THAM KHẢO**

*1* [*https://en.wikipedia.org/wiki/Principal\_component\_analysis*](https://en.wikipedia.org/wiki/Principal_component_analysis)

*2* [*http://www.cognotics.com/opencv/servo\_2007\_series/part\_2/index.html*](http://www.cognotics.com/opencv/servo_2007_series/part_2/index.html)

*3* [*http://www.bytefish.de/blog/pca\_in\_opencv/*](http://www.bytefish.de/blog/pca_in_opencv/)

*4* [*http://www.pages.drexel.edu/~nk752/tutorials.html*](http://www.pages.drexel.edu/~nk752/tutorials.html)

*5* [*http://dphtuan.uphero.com/dphtuannews/posts*](http://dphtuan.uphero.com/dphtuannews/posts)