TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

ffffffff

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**I**

BỘ MÔN KỸ THUẬT MÁY TÍNH

ĐỀ TÀI :

**NHẬN DẠNG BÀN TAY VÀ ĐẾM SỐ NGÓN TAY**

Sinh viên thực hiện: **Nguyễn Trọng Hảo**

Lớp CN\_CNTT1

Giáo viên hướng dẫn: ThS. **Nguyễn Đức Tiến**

HÀ NỘI 05 -2018

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

**1. Thông tin về sinh viên**

Họ và tên sinh viên: Nguyễn Trọng Hảo

Điện thoại liên lạc: 0983658905 Email:20166056@student.hust.edu.vn

Lớp: CN\_CNTT1 Hệ đào tạo: Đại học chính quy

Đồ án tốt nghiệp được thực hiện tại: Trường đại học Bách Khoa Hà Nội

**2. Mục đích nội dung của Đồ án 1**

* Tìm hiểu lập trình C# bằng winform và thư viện mã nguồn mở Emgu CV.
* Cách lưu trữ source code trên github và tải source code trên github về.
* Tìm hiểu cơ bản về xử lí ảnh thông qua không gian màu khác nhau.

**3. Các nhiệm vụ cụ thể của Đồ án 1**

* Tìm hiểu Github.
* Tìm hiểu về thư viện mã nguồn mở Emgu CV.
* Sử dụng các control trong winform như Textbox,listwiew,button… đưa vào chương trình .
* Kết nối Database
* Tổng kết và đánh giá.

**4. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của Đồ án 1 và cho phép bảo vệ:**

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 24 tháng 05 năm 2018*  Giáo viên hướng dẫn  *ThS Nguyễn Đức Tiến* |

**MỤC LỤC**

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 2](#_Toc294172297)

[MỤC LỤC 3](#_Toc294172302)

[A.NỘI DUNG 4](#_Toc294172303)

**PHẦN I : GIỚI THIỆU VỀ THƯ VIỆN EMGU CV…………………………4**

**1.Khái niệm Emgu CV………………………………………………………..4**

**2. Lợi thế của Emgu CV………………………………………………………4**

**3.Tích hợp vào.Net……………………………………………………………5**

[PHẦN II: TỔNG QUAN VỀ ĐỀ TÀI VÀ MỘT SỐ KHÁI NIỆM 7](#_Toc294172306)

[**1. Ý tưởng 7**](#_Toc294172307)

[**2. Một số khái niệm 9**](#_Toc294172315)

[PHẦN III: DEMO CHƯƠNG TRÌNH 12](#_Toc294172328)

[**1. Phân tích kết quả 12**](#_Toc294172329)

[**2. Một số hàm,control sử dụng trong chương trình 13**](#_Toc294172332)

[B.KẾT LUẬN 15](#_Toc294172339)

[TÀI LIỆU THAM KHẢO 15](#_Toc294172350)

**A.NỘI DUNG**

**PHẦN I : GIỚI THIỆU VỀ THƯ VIỆN EMGU CV**

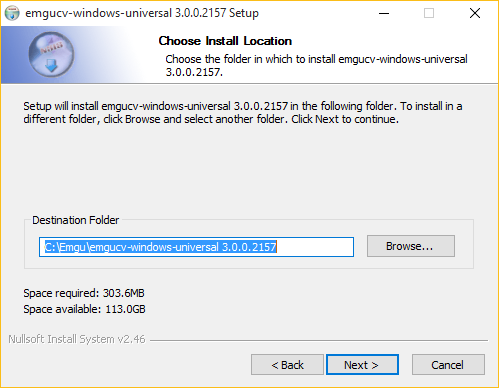
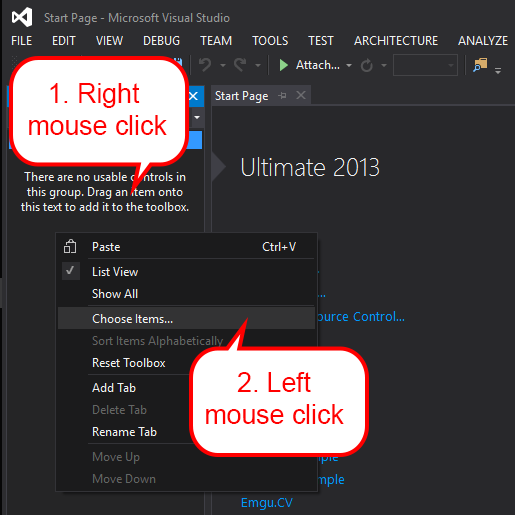
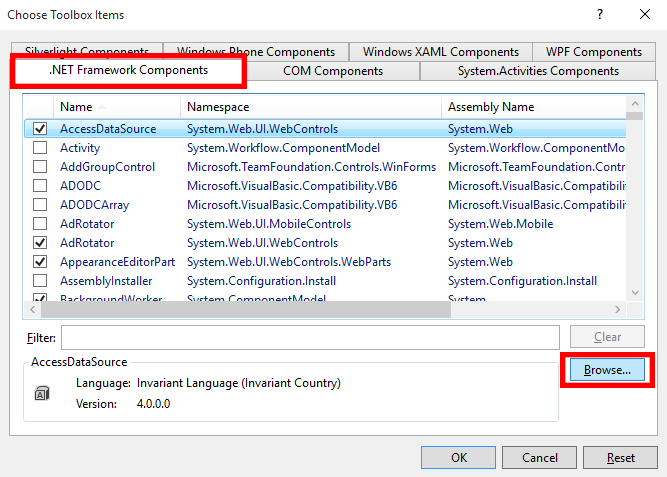
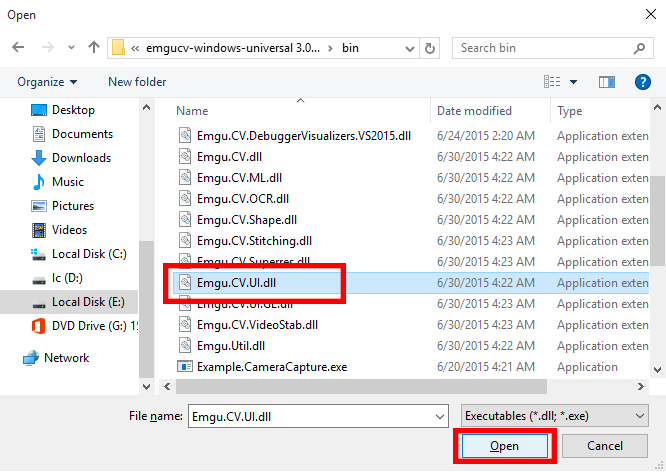
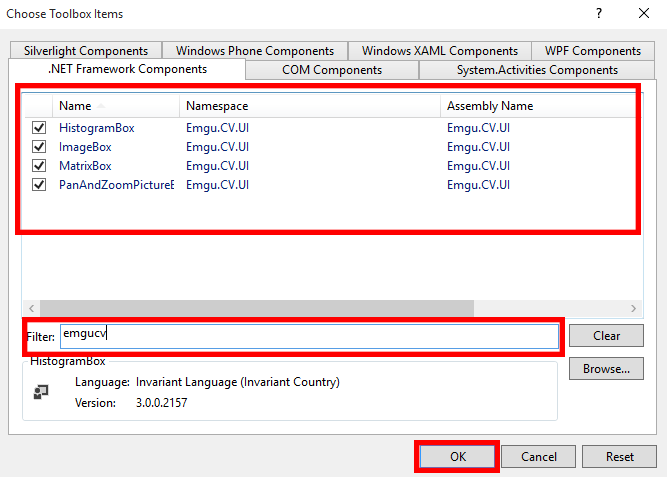
**1.Khái niệm Emgu CV**

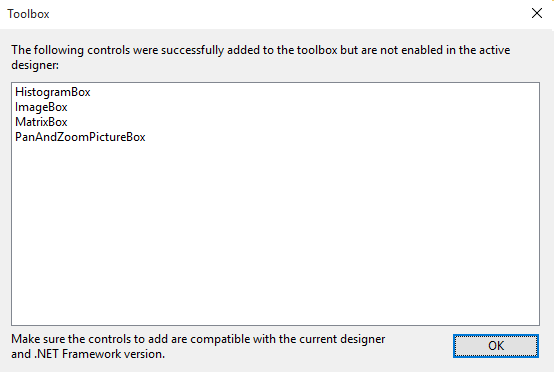
* Là một cross flatform .NET, một thư viện xử lý hình ảnh mạnh dành riêng cho ngôn ngữ C#. Cho phép gọi được chức năng của OpenCV  là từ .NET.
* Tương thích ngôn ngữ như: C#, VB, VC ++, Iron Python...
* Wrapper có thể được biên dịch bởi Visual Studio, Xamarin Studio và Unity.
* Nó có thể chạy trên Windows, Linux, Mac OS X, iOS, Android và Windows Phone.

**2.Lợi thế Emgu CV**

* EmguCV được viết hoàn toàn bằng C#. Có thể chạy trên bất kỳ nền tảng hỗ trợ bao gồm iOS, Android, Windows Phone, Hệ điều hành Mac OS X và Linux.
* EmguCV có thể được sử dụng từ nhiều ngôn ngữ khác nhau, bao gồm C#, VB.NET, C ++ và Iron Python.
* Nhận dạng ảnh: nhận dạng khuôn mặt, các vật thể …
* Xử lý ảnh: khử nhiễu, điều chỉnh độ sáng …
* Nhận dạng cử chỉ.
* Hỗ trợ tài liệu XML và intellisense.
* Sự lựa chọn để sử dụng hình ảnh lớp hoặc trực tiếp gọi chức năng từ OpenCV.

**3.Tích hợp vào.Net**

* EmguCV: Bạn có thể sử dụng bất kỳ version nào. Ở báo cáo này em sử dụng  EmguCV version 3.0.0. Link download : www.sourceforge.net/projects/emgucv/files/emgucv/
* Cài đặt EmguCV .Hãy chú ý tới các đường dẫn để đảm bảo rằng EmguCV sẽ hoạt động.
  + 
  + Thêm control của EmguCV vào Toolbox.
  + Hộp thoại **.NET Framwork Components**>**Browese...**
* 
* 2.Chọn tới thư mục chứa **EmguCV\emgucv-windows-universal**\_version**\bin\Emgu.CV.UI.dll >**
* 
* Trong **Filter:**gõ**emgucv**>Chọn tất cả các control hiện ra>**OK.**
* 
* Kết quả là trong toolbox sẽ có các control



# PHẦN II: TỔNG QUAN VỀ ĐỀ TÀI VÀ MỘT SỐ KHÁI NIỆM

1. **Ý tưởng** 
   * sử dụng phân đoạn màu da của con người để phát hiện đường viền của bàn tay kết hợp với sự phát hiện lồi tay để nhận ra cử chỉ và đếm số lượng ngón tay. Gồm 2 phần
   * Phần thứ nhất là 1 chương trình nhận biết da sử dụng 2 khung tham chiếu hình ảnh khác nhau .

* khung tham chiếu thực từ file nguồn(video file hoặc camera) với class YCrCbSkinDetector ().

Ảnh có chứa tường, trong nhà, người

Mô tả được tạo với mức tin cậy rất cao

* khung tham chiếu mặt na nhị phân với class HsvSkinDetector().
  + - 
* 1 thư viện gồm 1 số lớp để đếm số ngón tay dựa vào các khoảng không gian màu khác nhau . YrcCb và Hsv là 2 không gian màu khác nhau được sử dụng và được xác định bằng các giới hạn trên và dưới.( YrcCb\_min, YrcCb\_max,Hsv\_min,Hsv\_max trong source code) trong đó
* Convex hull : để xác định vỏ lồi bao bọc xung quanh bàn tay.Sử dụng phân đoạn màu da để xác định vỏ lồi này.
* Convexity defect : để nhận dạng chuyển động khi ta di chuyển bàn tay và đếm số ngon tay sử dụng 3 điểm : StartPoint,EndPoint và DepthPoint.
  + phần thứ 2 là chương trình đếm số ngón tay(FingerCount)sử dụng video hoặc camera gồm 3 chương trình con là camera,video và main.Khi thực hiện chương trình thì main sẽ gọi đến camera và video.
  + Cấu trúc của chương trình:

Ảnh có chứa ảnh chụp màn hình

Mô tả được tạo với mức tin cậy rất cao

1. **Một số khái niệm**
   * Các điểm ảnh cần lưu ý

Ảnh có chứa bản đồ, văn bản

Mô tả được tạo với mức tin cậy rất cao

* + Khởi tạo 3 điểm StartPoint,EndPoint,DepthPoint
* CircleF startCircle = new CircleF(startPoint, 5f);
* CircleF depthCircle = new CircleF(depthPoint, 5f);
* CircleF endCircle = new CircleF(endPoint, 5f);
  + Các câu lệnh vẽ lên các điểm StartPoint,EndPoint,DepthPoint
* currentFrame.Draw(startCircle, new Bgr(màu), size);
* currentFrame.Draw(depthCircle, new Bgr(màu), size);
* currentFrame.Draw(endCircle, new Bgr(màu), size);

* + Đường màu da cam là đường nối giữa startPoint và depthPoint để biểu hiện cho số ngón tay.Có bao nhiêu đường màu da cam thì sẽ có bấy nhiêu ngon tay được đếm.

Ảnh có chứa bản đồ

Mô tả được tạo với mức tin cậy rất cao

* + câu lệnh để vẽ đường nối giữa startPoint và depthPoint
* currentFrame.Draw(startDepthLine, new Bgr(Color.Orange), 2);
  + Đường màu canh lá cây là biggest contour biểu diễn cho đường viền bao bọc bàn tay.

Ảnh có chứa kẹp giấy, văn phòng phẩm

Mô tả được tạo với mức tin cậy rất cao

* + Sử dụng class private void ExtractContourAndHull(Image<Gray, byte> skin) để xác định đường này.
  + Sử dụng câu lệnh currentFrame.Draw(currentContour, new Bgr(Color.LimeGreen), 2) để vẽ nên đường này.

# PHẦN III: DEMO CHƯƠNG TRÌNH

1. **Phân tích kết quả** 
   * Giao diện khi nhấn nút Start :

Ảnh có chứa ảnh chụp màn hình

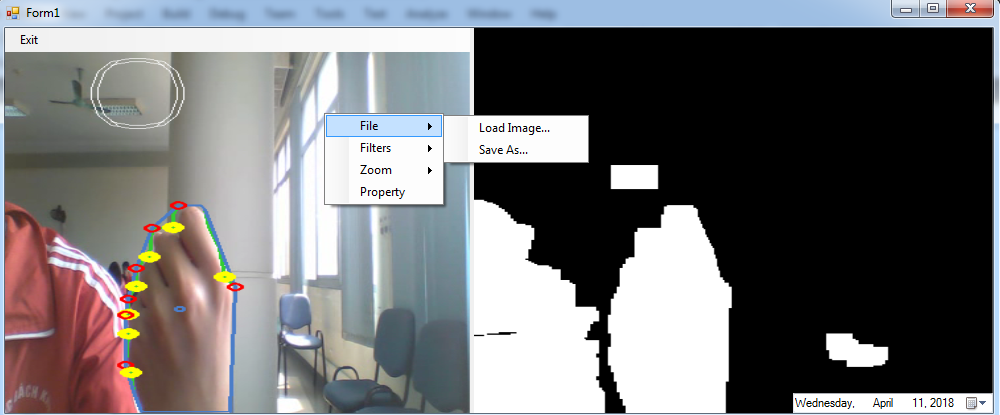
Mô tả được tạo với mức tin cậy rất cao

* + Click vào hình ảnh video nếu muốn chạy chương trình sử dụng video ,click vào hình ảnh camera nếu muốn chạy chương trình bằng camera
  + Khi thực hiện chạy chương trình thì trên màn hình sẽ xuất hiện giao diện sau

Ảnh có chứa trong nhà

Mô tả được tạo với mức tin cậy cao

* + Đây là 2 khung tham chiếu của bàn tay,phía bên tay trái là khung tham chiếu có được từ camera hoặc là video file, còn khung bên phải là mặt nạ nhị phân của nó.
  + Trên hình thì các điểm startpoint được biểu diễn bằng hình tròn màu đỏ, depthPoint biểu diễn bằng chấm màu đỏ và không có điểm endPoint.Đường màu xanh da trời bao bọc quanh bàn tay là convex Hull, đường màu da cam là đường nối giữa StartPoint và DepthPoint và đường màu xanh lá cây nhạt là contour.
  + Khi chúng ta di chuyển tay làm cho số ngón tay ít hơn hoặc nhiều hơn thì ngay lập tức đường màu da cam sẽ thay đổi tương ứng và số lượng ngón tay sẽ được hiển thị ở bên góc trái phía trên của màn hình(trong trường hợp này là 1).Đồng thời Convex Hull cũng thay đổi kích thước của nó so với ban đầu để phù hợp với trạng thái hiện tại của bàn tay.
  + Một số chức năng khác để xem các thuộc tính của kết quả



1. **Một số hàm,control sử dụng trong chương trình**
   * PointF(single,single) để khởi tạo 1 điểm với các tọa độ cụ thể
   * LineSegment2D(Point2D<T>, Point2D<T>) để tạo 1 đoạn thẳng với điểm đầu và điểm cuối cụ thể
   * CircleF(PointF, Single) để tạo 1 vòng tròn với bán kính cụ thể
   * MCvFont(FONT, Double, Double) để tạo phông chữ (hiển thị số lượng ngón tay) với kích thước tự tạo.
   * Một số control sử dụng : MenuStrip , imageBox , DateTimePicker , Button ,ToolTip , ContextMenuStrip ,Label

# B.KẾT LUẬN

- Chương trình đã hoàn thành một số tính năng chính nhưng vẫn còn nhiều điểm khuyết thiếu :

* + - * Chất lượng video file và camera không thật sự tốt,còn phụ thuộc nhiều vào chất lượng ánh sáng,góc quay.
      * Giao diện chưa được tốt.
      * Khi chaỵ chương trình camera trước thì khi chạy chương trình sử dụng video sẽ chậm hơn so với bình thường.
      * Số lượng control vẫn đang còn ít.
      * Chưa kết nối được với database.

- Các giải pháp :

* + - * Tìm hiểu về xử lí ảnh trong C# để có thể thay đổi toạ độ theo mong muốn
      * Tìm hiểu thêm về cách thiết kế giao diện trên winform

# TÀI LIỆU THAM KHẢO

1. <https://www.youtube.com/watch?v=Fjj9gqTCTfc&t=5s>
2. <http://anikettatipamula.blogspot.ro/2012/02/hand-gesture-using-opencv.html>