**TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM  
KHOA CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**

****

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN “XỬ LÝ ẢNH”**

***Đề tài:***

***NHẬN DẠNG SỐ CĂN CƯỚC CÔNG DÂN***

***TỪ ẢNH THẺ CĂN CƯỚC CÔNG DÂN***

***GV hướng dẫn : Nguyễn Hữu Tuân***

***Sinh viên thực hiện: Nguyễn Thế Anh – Mã SV: 86271***

***Hải Phòng, tháng 5 năm 2023***

**TRƯỜNG ĐẠI HỌC HÀNG HẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN HỆ THỐNG THÔNG TIN**

**-----\*\*\*-----**

**BÀI TẬP LỚN**

**HỌC PHẦN: “*XỬ LÝ ẢNH*”**

1. **Tên đề tài**

Nhận dạng số CCCD từ ảnh thẻ CCCD.

1. **Mục đích**

Thiết kế và xây dựng chương trình nhận dạng số CCCD từ ảnh thẻ CCCD sử dụng thư viện OpenCV.

1. **Công việc cần thực hiện**

* Tìm hiều bài thuật toán nhận dạng số CCCD từ ảnh thẻ CCCD.
* Tìm hiểu các thư viện và công cụ hỗ trợ.
* Xây dựng chương trình nhận dạng số CCCD từ ảnh thẻ CCCD và kiểm thử.
* Bảo vệ bài tập lớn.

1. **Yêu cầu**

* Kết quả làm bài tập lớn: Báo cáo bài tập lớn.
* Báo cáo bài tập lớn phải được trình bày theo mẫu quy định (kèm theo).
* Hạn nộp bài tập lớn: thứ 2, tuần thứ 14.

1. **Tài liệu tham khảo**

* Sách giáo trình Xử lý ảnh**,** khoa CNTT, ĐH HH VN.
* voicegpt.us

MỤC LỤC

[CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN 1](#_Toc134446662)

[1.1. ĐẶT VẤN ĐỀ 1](#_Toc134446663)

[1.2. MÔ TẢ BÀI TOÁN 2](#_Toc134446664)

[CHƯƠNG II. THUẬT TOÁN 3](#_Toc134446665)

[2.1 Các bước thực hiện 3](#_Toc134446666)

[2.2 Các thuật toán được sử dụng 3](#_Toc134446667)

[CHƯƠNG III. CÀI ĐẶT 7](#_Toc134446668)

[3.1. Ngôn ngữ lập trình python 7](#_Toc134446669)

[3.2. IDE Pycharm 7](#_Toc134446670)

[3.3. Thư viện 8](#_Toc134446671)

[a. Opencv 8](#_Toc134446672)

[b. Pytesseract 9](#_Toc134446673)

[3.4. Cài đặt chương trình 11](#_Toc134446674)

[CHƯƠNG IV. THỬ NGHIỆM 13](#_Toc134446675)

[KẾT LUẬN 17](#_Toc134446676)

[TÀI LIỆU THAM KHẢO 18](#_Toc134446677)

**LỜI MỞ ĐẦU**

Nhận dạng số CCCD từ ảnh thẻ CCCD là một trong những ứng dụng có phần mới của trí tuệ nhân tạo tại Việt Nam. Nó được sử dụng trong nhiều lĩnh vực như các thủ tục hành chính, hợp đồng, quản lý dữ liệu dân cư và nhiều ứng dụng khác. Tuy nhiên, việc nhận dạng số CCCD từ ảnh thẻ CCCD vẫn đang là một thách thức lớn đối với các nhà nghiên cứu và chuyên gia trong lĩnh vực này. Báo cáo này sẽ giới thiệu về các phương pháp, kỹ thuật và công nghệ được sử dụng để nhận dạng số CCCD từ ảnh thẻ CCCD, cũng như những ứng dụng thực tế của chúng. Ngoài ra, báo cáo cũng sẽ đề cập đến các thách thức và hạn chế của công nghệ nhận dạng số CCCD từ ảnh thẻ CCCD, cùng với những nỗ lực để cải thiện và phát triển công nghệ này trong tương lai.

# CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN

## ĐẶT VẤN ĐỀ

Nhận dạng số CCCD từ ảnh thẻ CCCD là một trong những công nghệ đang được phát triển nhanh chóng trong thời gian gần đây. Tuy nhiên, những tiềm năng và nguy cơ của công nghệ này đang trở thành đề tài tranh cãi và đặt ra nhiều thách thức đối với xã hội ngày nay.

Một trong những vấn đề đối với xã hội là việc sử dụng nhận dạng số CCCD từ ảnh thẻ CCCD trong lĩnh vực an ninh và giám sát. Mặc dù công nghệ này có thể giúp cho các cơ quan chức năng tiết kiệm thời gian và tăng độ chính xác, nhưng việc sử dụng quá mức và trái phép có thể gây ra sự xâm phạm đến quyền riêng tư của công dân. Nhiều người lo ngại rằng việc thu thập và lưu trữ dữ liệu số CCCD có thể dẫn đến sự lạm dụng và lộ ra thông tin cá nhân của họ.

Bên cạnh đó, công nghệ nhận dạng số CCCD từ ảnh thẻ CCCD cũng đang gặp phải nhiều thách thức về tính chính xác và độ tin cậy. Các hệ thống nhận dạng số CCCD từ ảnh thẻ CCCD hiện nay vẫn chưa hoàn toàn đáp ứng được các yêu cầu của thực tiễn, đặc biệt là trong các trường hợp phức tạp như khi thẻ CCCD bị mờ, hỏng, mất số...

Trong bối cảnh đó, việc nghiên cứu và phát triển công nghệ nhận dạng số CCCD từ ảnh thẻ CCCD có tính chính xác và độ tin cậy cao, đồng thời đảm bảo đúng quyền riêng tư và an toàn thông tin của người dùng, là cực kỳ cần thiết.

* **Các ưu điểm của nhận dạng số CCCD từ ảnh thẻ CCCD :**

Tiết kiệm thời gian và công sức: Nhận dạng số CCCD từ ảnh thẻ CCCD có thể tự động hoá quá trình quét thẻ lấy số của một người mà không cần phải thực hiện bằng tay, tiết kiệm thời gian và công sức.

Độ chính xác cao: Các thuật toán nhận dạng số CCCD từ ảnh thẻ CCCD được phát triển trong thời gian qua đã đạt được độ chính xác cao, giúp giảm thiểu sai sót và tăng độ tin cậy của quá trình nhận diện.

* **Các nhược điểm của nhận dạng số CCCD từ ảnh thẻ CCCD:**

Ảnh hưởng đến quyền riêng tư: Nhận dạng số CCCD từ ảnh thẻ CCCD có thể dẫn đến việc thu thập dữ liệu cá nhân mà không được sự đồng ý của người dùng, ảnh hưởng đến quyền riêng tư của họ.

Độ chính xác bị ảnh hưởng bởi nhiều yếu tố: Nhận dạng số CCCD từ ảnh thẻ CCCD có thể bị ảnh hưởng bởi nhiều yếu tố khác nhau, bao gồm ánh sáng, góc chụp, chất lượng thẻ và chữ số trên thẻ, dẫn đến độ chính xác thấp.

Điều chỉnh sai lệch: Nếu hệ thống nhận dạng số CCCD từ ảnh thẻ CCCD bị điều chỉnh sai lệch, nó có thể dẫn đến việc phát hiện sai số CCCD.

## 1.2. MÔ TẢ BÀI TOÁN

Đầu tiên chương trình sẽ mở cửa sổ camera để thu thập ảnh thẻ CCCD. Sau đó chương trình sẽ tiến hành xử lý ảnh nhận được. Cuối cùng chương trình sẽ hiển thị kết quả là 12 số của dãy số CCCD, nếu không nhận dạng được, chương trình sẽ trả lại về kết quả unknow.

# CHƯƠNG II. THUẬT TOÁN

## 2.1 Các bước thực hiện

Để thực hiện tốt bài toán, trước hết, chúng ta cần có sơ đồ xây dựng các bước thực hiện. Sơ đồ các bước thực hiện bài toán đọc số CCCD từ ảnh thẻ CCCD:



Hình 2. 1 Sơ đồ các bước thực hiện

## 2.2 Các thuật toán được sử dụng

Sau khi vẽ sơ đồ các bước thực hiện, chúng ta sẽ tìm hiểu thuật toán áp dụng đối với từng bước.

* **Bước 1. Kết nối camera**

Chương trình sử dụng cv2.VideoCapture(0) để tạo đối tượng VideoCapture và lấy ảnh từ camera web được kết nối với máy tính. Đối tượng VideoCapture được sử dụng để đọc các khung ảnh từ camera và truyền chúng cho các bước tiếp theo để xử lý. Hàm cap.read() để đọc ảnh từ webcam qua đối tượng VideoCapture vừa được tạo, trả về hai giá trị là ret và frame. Lấy ảnh và đọc ảnh khác nhau ở chỗ đọc ảnh chỉ thực hiện trên ảnh được chuyển đổi sang ảnh xám trong khi lấy ảnh sẽ thực hiện trên tất cả các ảnh chụp liên tiếp từ camera, bao gồm cả ảnh màu và ảnh xám.

* **Bước 2. Đổi màu hình ảnh**

Sau khi đã đọc được ảnh, sẽ chuyển đổi không gian màu của ảnh từ BGR sang xám. Chuyển đổi không gian màu từ BGR sang xám nhằm mục đích giảm chiều dài vector màu (3 thành phần màu được đánh giá riêng biệt) thành một ma trận đơn giản chứa mức xám duy nhất tại mỗi pixel. Điều này giúp giảm độ phức tạp khi xử lý hình ảnh, đặc biệt là trong các bước tiếp theo.

* **Bước 3. Giảm nhiễu**

Sau đó, sử dụng bộ lọc Gaussian qua thuật toán *GaussianBlur* để lọc nhiễu. Làm mờ và giảm nhiễu hình ảnh nhằm loại bỏ các nhiễu nhỏ và giảm thiểu sự ảnh hưởng của các nhiễu lớn trên hình ảnh, giúp các đối tượng được phân biệt rõ ràng hơn trong các bước tiếp theo.

* **Bước 4. Chuyển ngưỡng để tách kí tự**

Sau khi giảm nhiễu, tạo bản đồ nhị phân bằng cách sử dụng các kỹ thuật chuyển đổi ngưỡng như *cv2.threshold()* để tạo ra một hình ảnh chỉ gồm hai giá trị pixel có giá trị 0 hoặc 255. Bản đồ nhị phân giúp phân tách đối tượng từ nền của hình ảnh . Thuật toán này thực hiện tách ký tự bằng cách chuyển đổi ngưỡng tự động dựa trên phương sai tối thiểu hai lớp giá trị pixcel được xác định bằng phương pháp dò tìm ngưỡng Otsu.

* **Bước 5. Xác định vùng chứa số CCCD**

Chương trình xác định vùng chứa số CCCD bằng cách sử dụng các biến x, y, z, h để xác định các toạn độ, kích thước của hình chữ nhật cần lấy từ ảnh đã được chuyển đổi trước đó. Vùng này được xác định là một phần của ảnh ở tọa độ (x, y) và có chiều rộng w, chiều cao h, nhằm đảm bảo phần lấy sẽ nằm trọn vẹn trong số CCCD.

* **Bước 6. Nhận dạng số CCCD**

Chương trình sử dụng thư viện pytesseract để nhận dạng số từ vùng ảnh chứa số CCCD được xác định từ bước 5. Hàm image\_to\_string() được sử dụng của pytesseract sử dụung các kĩ thuật xử lý ngôn ngữ tự nhiên để tìm và trích xuất các ký tự có thể đọc được từ trong ảnh, từ đó trả về một chuỗi đại diện cho số CCCD. Việc này được thực hiện dựa trên mô hình nhận diện ký tự và ảnh học ( machien learning) được tích hợp sẵn trong thư viện pytesseract.

* **Bước 7. Loại bỏ ký tự không phải là số**

Chương trình sử dụng hàm filter() để loại bỏ bất kỳ ký tự nào không phải là số trong chuỗi được nhận dạng từ hình ảnh đại diện cho số CCCD bằng thư viện pytesseract. Hàm filter() nhận đầu vào là một hàm lamdba và một list các ký tự đại diện cho chuỗi cần xử lý, sau đó chỉ giữ lại các ký tự là số. Việc loại bỏ các ký tự không phải số giúp tăng độ chính xác cho kế quả nhận dạng được.

* **Bước 8. Kiểm tra điều kiện**

Vì số CCCD là dãy gồm đúng 12 chữ số nên sử dụng hàm if len() để kiểm tra, nếu thỏa mãn sẽ dùng hàm print() để hiển thị kết quả là số CCCD, nếu không thỏa mãn, không hiển thị gì.

* **Bước 9. Hiển thị ảnh**

Hình ảnh `frame` là hình ảnh BGR ban đầu, được sử dụng để hiển thị đầy đủ khung hình đang được chụp từ camera. Hình ảnh `thresh` là hình ảnh nhị phân sau khi đã được xử lý và được hiển thị để thể hiện chỉ rõ các vùng màu đen và trắng tương ứng với các đối tượng và nền. Hai khung hình đều được hiển thị trong cửa sổ trực quan để theo dõi quá trình xử lý ảnh và kiểm tra kết quả của thuật toán..

* **Bước 10. Thoát chương trình và giải phóng tài nguyên**

Để giải phóng tài nguyên, sử dụng `cap.release()` để giải phóng camera và thả bộ nhớ được sử dụng bởi chương trình. Sau đó, sử dụng `cv2.destroyAllWindows()` để đóng tất cả các cửa sổ hiển thị và giải phóng tài nguyên được sử dụng bởi các cửa sổ đó. Việc giải phóng tài nguyên này giúp giảm thiểu nguy cơ tiêu tốn bộ nhớ trong quá trình sử dụng, đồng thời giúp giải phóng tài nguyên cho phần cứng và chương trình khác trên máy tính.

# CHƯƠNG III. CÀI ĐẶT

## 3.1. Ngôn ngữ lập trình python

Python là ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học. Các nhà phát triển sử dụng python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau.

Python thường được phát triển trong nhiều lĩnh vực , VD:

* + Phát triển web phía máy chủ.
  + Tự động hóa bằng các tập lệnh python.
  + Khoa học dữ liệu và máy học.
  + Phát triển phần mềm.
  + Tự động hóa kiểm thử phần mềm.

Cài đặt python: tải xuống và cài đặt python theo đường dẫn: <https://www.python.org/downloads/> .

Kiểm tra phiên bản python hiện tại trên máy bằng cách chạy lệnh cmd:

* + Bật Command Prompt bằng cách ấn tổ hợp ‘Windows + R’ để mở hộp thoại “Run”, gõ “cmd” và nhấn Enter.
  + Sau đó chạy lệnh *python –version.*
  + Nếu máy chưa cài đặt python, lệnh này sẽ không thành công1

## 3.2. IDE Pycharm

**IDE là gì?** – **IDE viết tắt là từ** (**I**ntegrated **D**evelopment **E**nvironment) là môi trường tích hợp dùng để viết code để phát triển ứng dụng. Ngoài ra **IDE** tích hợp các tool hỗ trợ khác như trình biên dịch (**Compiler**), trình thông dịch (**Interpreter**), kiểm tra lỗi (**Debugger**), định dạng hoặc highlight code, tổ chức thư mục code, tìm kiếm code…

Các môi trường IDE bao gồm :

* + Một trình soạn thảo mã nguồn (*source code editor*): dùng để viết mã.
  + Trình biên dịch (*compiler*) và/hoặc trình thông dịch (*interpreter*).
  + Công cụ xây dựng tự động: khi sử dụng sẽ biên dịch (hoặc thông dịch) mã nguồn, thực hiện liên kết (*linking*), và có thể chạy chương trình một cách tự động.
  + Trình gỡ lỗi (*debugger*): hỗ trợ dò tìm lỗi.
  + Ngoài ra, còn có thể bao gồm hệ thống quản lý phiên bản và các công cụ nhằm đơn giản hóa công việc xây dựng giao diện người dùng đồ họa (GUI).

PyCharm là môi trường phát triển tích hợp đa nền tảng (IDE) được phát triển bởi Jet Brains và được thiết kế đặc biệt cho Python. PyCharm có mặt trên cả 3 nền tảng Windows, Linux và Mac OS.

Pycharm là một giải pháp phù hợp cho Python developers vì IDE này hỗ trợ nhiều extensions, môi trường ảo (Virtual Environment), nhiều tính năng thông minh như bộ code completion, tự động thụt lề, phát hiện văn bản trùng lặp và kiểm tra lỗi. Ngoài ra còn có các tính năng tìm kiếm mã nguồn thông minh để tìm kiếm từng từ một trong nháy mắt.

Cài đặt Pycharm : tải và cài đặt pycharm theo đường dẫn : <https://www.jetbrains.com/pycharm/download/>

## 3.3. Thư viện

### a. Opencv

OpenCV (Open Source Computer Vision Library) là một thư viện mã nguồn mở chuyên về xử lý ảnh và thị giác máy tính. Thư viện này được phát triển bởi Intel vào năm 1999 và hiện nay đã trở thành một trong những thư viện xử lý ảnh phổ biến nhất trên thế giới, được sử dụng rộng rãi trong các ứng dụng như nhận dạng khuôn mặt, phân loại ảnh, phát hiện đối tượng, chuyển đổi màu sắc, lọc ảnh và nhiều ứng dụng khác.

OpenCV cung cấp các công cụ và hàm xử lý ảnh, cho phép người dùng thao tác trên các định dạng ảnh khác nhau như BMP, JPEG, PNG, TIFF, v.v. OpenCV cũng hỗ trợ việc xử lý video và webcam.

OpenCV có thể được sử dụng trong các ngôn ngữ lập trình như C++, Python, Java và MATLAB, cho phép các nhà phát triển lập trình các ứng dụng xử lý ảnh đa nền tảng và đa ngôn ngữ. Thư viện này còn có thể được cài đặt trên các hệ điều hành khác nhau như Windows, Linux, macOS và Android.

OpenCV là một trong những công cụ quan trọng trong lĩnh vực thị giác máy tính và xử lý ảnh, đóng vai trò quan trọng trong nhiều lĩnh vực ứng dụng, từ công nghiệp đến y tế và giải trí.

Để sử dụng thư viện Opencv, chúng ta cần cài đặt nó trước. Cài đặt Opencv bằng cách sau : Mở Pycharm => File => Setting => Chọn Project => Python Interprerter => Dấu cộng (install) => Tìm và cài đặt thư viện opencv-python . Khi đã cài đặt thành công, chương trình sẽ báo *Package ‘openvc-python’ installed successfully* .

### b. Pytesseract

Pytesseract là một thư viện mã nguồn mở dùng để nhận dạng ký tự quang học (OCR) trong Python. Thư viện này kết nối với Tesseract, một công cụ OCR mã nguồn mở được phát triển bởi Google.

Các tính năng chính của Pytesseract là:

Nhận dạng ký tự quang học từ ảnh hoặc tệp PDF

Có thể nhận dạng các ngôn ngữ khác nhau

Thuận tiện cho các ứng dụng OCR với công suất thấp

Để sử dụng Pytesseract trong Python, ta có thể cài đặt thư viện Pytesseract và Tesseract bằng pip. Sau khi cài đặt xong, ta có thể sử dụng các hàm của thư viện để thực hiện nhận dạng ký tự trên tệp ảnh hoặc tệp PDF.

Các phương thức API cơ bản của Pytesseract gồm:

`pytesseract.image\_to\_string(image, lang=None, config=None)` : Phương thức này dùng để nhận dạng các ký tự trên ảnh hoặc PDF. Trong đó, `image` là tệp ảnh được đọc bằng Python Imaging Library (PIL), `lang` là mã ISO 639-1 cho ngôn ngữ được sử dụng, `config` là các tham số điều chỉnh thêm cho Tesseract.

`pytesseract.image\_to\_boxes(image, lang=None, config=None)` : Phương thức này trả về một danh sách các ô vuông bao quanh các ký tự được nhận dạng. Các ô vuông này bao gồm thông tin về toạ độ của các ký tự, giống như một hộp giới hạn.

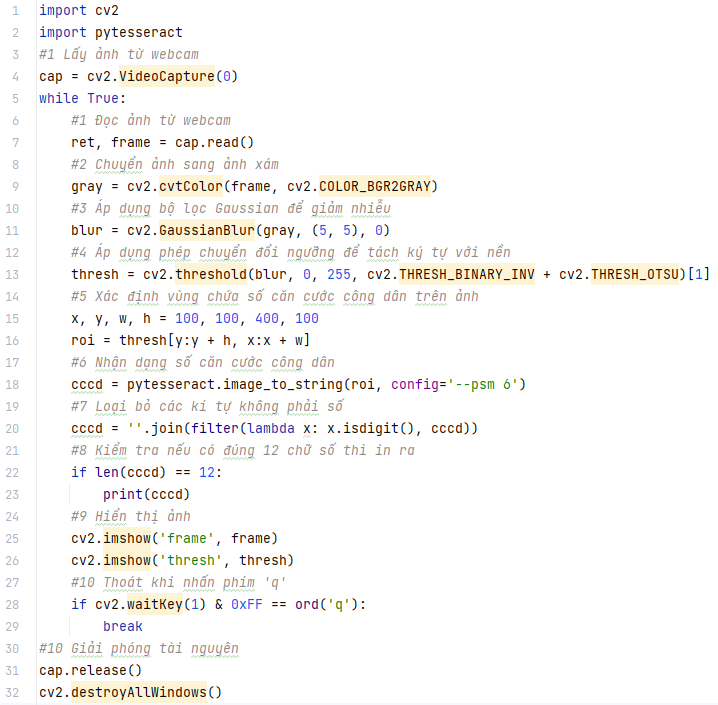
`pytesseract.image\_to\_data(image, lang=None, config=None)` : Phương thức này trả về một danh sách các từ, cùng với toạ độ của chúng trên ảnh, độ tin cậy của việc nhận dạng từ đó, và thông tin về vùng chứa từ đó trên ảnh.

Pytesseract được sử dụng phổ biến trong các ứng dụng nhận dạng ký tự trên tệp ảnh hoặc PDF, nó rất hữu hiệu cho việc nhận diện chữ viết tay, chữ in và số liệu.

Để sử dụng thư viện pytesseract ,chúng ta cần cài đặt nó trước. Cài đặt Opencv bằng cách sau : Mở Pycharm => File => Setting => Chọn Project => Python Interprerter => Dấu cộng (install) => Tìm và cài đặt thư viện opencv-python . Khi đã cài đặt thành công, chương trình sẽ báo *Package ‘openvc-python’ installed successfully* .

## 3.4. Cài đặt chương trình

Code chương trình trên IDE Pycharm:



Code chương trình cụ thể :

import cv2  
import pytesseract  
  
*#1 Lấy ảnh từ webcam*cap = cv2.VideoCapture(0)  
  
while True:  
 *#2 Đọc ảnh từ webcam* ret, frame = cap.read()  
  
 *#3 Chuyển ảnh sang ảnh xám* gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)  
  
 *#4 Áp dụng bộ lọc Gaussian để giảm nhiễu* blur = cv2.GaussianBlur(gray, (5, 5), 0)  
  
 *#5 Áp dụng phép chuyển đổi ngưỡng để tách ký tự với nền* thresh = cv2.threshold(blur, 0, 255, cv2.THRESH\_BINARY\_INV + cv2.THRESH\_OTSU)[1]  
  
 *#6 Xác định vùng chứa số căn cước công dân trên ảnh* x, y, w, h = 100, 100, 400, 100  
 roi = thresh[y:y + h, x:x + w]  
  
 *#7 Nhận dạng số căn cước công dân* cccd = pytesseract.image\_to\_string(roi, config='--psm 6')  
  
 *#8 Loại bỏ các kí tự không phải số* cccd = ''.join(filter(lambda x: x.isdigit(), cccd))  
  
 *#9 Kiểm tra nếu có đúng 12 chữ số thì in ra* if len(cccd) == 12:  
 print(cccd)  
  
 *#10 Hiển thị ảnh* cv2.imshow('frame', frame)  
 cv2.imshow('thresh', thresh)  
  
 *#11 Thoát khi nhấn phím 'q'* if cv2.waitKey(1) & 0xFF == ord('q'):  
 break  
9  
*#12 Giải phóng tài nguyên*cap.release()  
cv2.destroyAllWindows()

# CHƯƠNG IV. THỬ NGHIỆM

Kết quả chạy thử nghiệm chương trình :

Tổng quát :

Chi tiết :

+ Ảnh ban đầu :

+ Ảnh nhị phân :

Kết quả thu được :

Với ảnh thẻ có độ nét cao, độ tương phản tốt và khoảng cách từ ảnh thẻ tới webcam đủ gần, chương trình đã cho kết quả chính xác.

Số CCCD trên thẻ CCCD :

Số CCCD chương trình đọc được :

# KẾT LUẬN

Sau một thời gian tìm hiểu, nghiên cứu, được sử hướng dẫn nhiệt tình của thầy Nguyễn Hữu Tuân, em đã hoàn thành được bài tập lớn. Qua bài tập lớn, em đã có điều kiện được hiểu sâu hơn về lĩnh vực về xử lý ảnh với Python và thư viện Opencv, cũng như qua môn học bọn em được thực hành và tiếp xúc trực tiếp. Chương trình trên có một số ưu điểm và nhược điểm :

* + Ưu điểm: khi điều kiện môi trường tốt: độ sáng phù hợp, khoảng cách từ webcam tới ảnh thẻ đủ gần và chất lượng ảnh thẻ tốt : sáng sủa, rõ ràng, chương trình cho kết quả rất chính xác .
  + Nhược điểm: chương trình đòi hỏi điều kiện khá cao để nhận dạng, trong điều kiện bình thường, chương trình lúc nhận dạng được lúc không, kết quả nhận dạng lúc đúng lúc sai, chưa được tối ưu.

# TÀI LIỆU THAM KHẢO

1. Giáo trình học phần “ Xử lý ảnh”, Trường Đại học Hàng hải Việt Nam.

2. Thông tin Python tại : <https://aws.amazon.com/vi/what-is/python/> .

3. Thông tin IDE Pycharm tại : <https://t3h.edu.vn/tin-tuc/pycharm-la-gi-huong-dan-cai-dat-va-su-dung-pycharm> và <https://codelearn.io/sharing/pycharm-ide-cho-lap-trinh-python> .

4. Thông tin thư viện Openvc tại : <https://topdev.vn/blog/opencv-la-gi-hoc-computer-vision-khong-kho/> .

5. Thư viện Pytesseract tại : <https://thigiacmaytinh.com/nhan-dien-van-ban-bang-tesseract> .

6. AI hỗ trợ : <https://voicegpt.us/chatgpt> .