

**TRƯỜNG ĐẠI HỌC BÁCH KHOA TP HCM**

**KHOA ĐIỆN – ĐIỆN TỬ**

**Bộ môn Viễn Thông**



**LUẬN VĂN TỐT NGHIỆP**

# **HỆ THỐNG NHẬP ĐIỂM TỰ ĐỘNG TỪ ẢNH BÀI THI**

*Hướng dẫn:* **ThS. ĐẶNG NGỌC HẠNH**

*Sinh viên thực hiện:* **MAI CHÍ BẢO**

**MSSV: 1710586**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

Số: \_\_\_\_\_/BKĐT

Khoa: **Điện – Điện tử**

Bộ Môn: **Viễn Thông**

## NHIỆM VỤ LUẬN VĂN TỐT NGHIỆP

- Họ và tên: Mai Chí Bảo MSSV: 1710586
  - Ngành: Điện – Điện tử Chuyên ngành: Kỹ thuật Điện tử - Truyền thông
  - Đề tài: Hệ thống nhập điểm tự động từ ảnh bài thi
  - Nhiệm vụ:
    - Thiết kế hệ thống nhập điểm tự động từ ảnh bài thi
    - Trích tách, nhận dạng thông tin với mô hình CRNN kết hợp CTC Loss
    - Tạo ra giao diện tương tác với người dùng
    - Phân tích và so sánh kết quả nhận dạng
  - Ngày giao nhiệm vụ luận văn: 20/12/2020
  - Ngày hoàn thành nhiệm vụ: 23/12/2021
  - Họ và tên người hướng dẫn:
    - ThS. Đặng Ngọc Hạnh
- BM Viễn Thông, Khoa Điện – Điện Tử
- Nội dung và yêu cầu LVTN đã được thông qua Bộ Môn.

Phản hướng dẫn  
100%

TP.HCM, ngày ... tháng ... năm 2022

**CHỦ NHIỆM BỘ MÔN**

**NGƯỜI HƯỚNG DẪN CHÍNH**

PGS. TS. Hà Hoàng Kha

ThS. Đặng Ngọc Hạnh

### PHẦN DÀNH CHO KHOA, BỘ MÔN:

Người duyệt (chấm sơ bộ): .....

Đơn vị: .....

Ngày bảo vệ: .....

Điểm tổng kết: .....

Nơi lưu trữ luận văn: .....

## LỜI CẢM ƠN

Lời đầu tiên em xin chân thành cảm ơn quý thầy, cô trường Đại Học Bách Khoa Thành Phố Hồ Chí Minh đã tận tình dạy dỗ em trong suốt những năm học vừa qua. Trong đó phải kể đến công lao to lớn của thầy, cô Khoa Điện – Điện tử đã tạo điều kiện cho em được học tập và hoàn thành luận văn tốt nghiệp này.

Trong quá trình tìm hiểu và nghiên cứu, có rất nhiều kiến thức cần phải chất lọc và em không hiểu hết chúng. Xin cảm ơn cô Đặng Ngọc Hạnh đã chỉ bảo em phải biết tập trung vào kiến thức gì là tốt và cô đã tận tình giúp em chỉnh sửa luận văn này. Em xin cảm ơn thầy Phạm Việt Cường đã truyền cảm hứng và giảng dạy một cách chi tiết về nhiều kiến thức quan trọng trong môn Trí Tuệ Nhân Tạo và Thị Giác Máy.

Cám ơn ba mẹ đã là nơi động viên tinh thần rất lớn cho con. Cám ơn em gái đã hỗ trợ thiết kế giấy thi mẫu của trường Bách Khoa trong thời gian giãn cách xã hội. Cám ơn bạn bè đã hỗ trợ mình tạo tập dữ liệu phục vụ cho luận văn này. Xin cảm ơn câu lạc bộ BKMC – Bách Khoa Music Club đã cùng mình có rất nhiều kỉ niệm tươi đẹp thời sinh viên.

Với thời gian thực hiện luận văn ngắn và chưa có nhiều kinh nghiệm thực tiễn, luận văn này chắc chắn sẽ có nhiều thiếu sót. Em rất mong sự góp ý từ thầy, cô để đề tài trở nên hoàn thiện và có thể trong tương lai không xa sẽ được áp dụng vào thực tế, góp phần giải tỏa áp lực cho hàng triệu giáo viên.

*TP. Hồ Chí Minh, ngày 23 tháng 12 năm 2021*

*Sinh viên thực hiện luận văn*

*Mai Chí Bảo*

## LỜI CAM ĐOAN

Em tên: Mai Chí Bảo, là sinh viên chuyên ngành Kỹ thuật Điện tử – Truyền thông, niên khóa 2017 – 2022, tại Đại học Quốc gia thành phố Hồ Chí Minh – Trường Đại học Bách Khoa. Em xin cam đoan những nội dung sau đều là sự thật: (i) Công trình nghiên cứu này hoàn toàn do chính em thực hiện. (ii) Các tài liệu và trích dẫn trong luận văn này được tham khảo từ các nguồn thực tế, có uy tín và độ chính xác cao. (iii) Các số liệu và kết quả của công trình này được em tự thực hiện một cách độc lập và trung thực.

*TP. Hồ Chí Minh, ngày 23 tháng 12 năm 2021*

*Sinh viên thực hiện luận văn*

*Mai Chí Bảo*

## TÓM TẮT LUẬN VĂN

Thị giác máy là một trong những lĩnh vực khá mới mẻ và hứa hẹn nhiều bước phát triển nhảy vọt trong tương lai và sẽ đóng góp cho sự phát triển của hầu hết tất cả các lĩnh vực. Tuy nhiên với giáo dục, một lĩnh vực có ảnh hưởng rất lớn đến thế hệ mai sau, vẫn chưa được đầu tư đúng mức. Ngoài những giờ dạy và học trên giảng đường, lớp học. Giáo viên vẫn phải bỏ ra quá nhiều thời gian không liên quan đến việc giáo dục, giảng dạy kiến thức. Cụ thể là việc nhập điểm số.

Đề tài: “Hệ thống nhập điểm tự động từ ảnh bài thi” được thực hiện dựa trên ý tưởng đó. Đây là một đề tài rất có ý nghĩa trong việc giảm tải lượng công việc của giáo viên, giúp họ tập trung thời gian, sức khỏe vào những công việc liên quan đến giáo dục nhiều hơn.

Trong đề tài này, ngoài những kiến thức về xử lý ảnh, đề tài còn tập trung đi sâu nghiên cứu về nhận dạng chữ viết tay bằng mô hình CRNN kết hợp CTC Loss và các bước huấn luyện nhằm tăng khả năng nhận dạng.

Hệ thống sử dụng Camera điện thoại để thu nhận hình ảnh, truyền về cho máy tính xử lý, nhận dạng thông tin sinh viên trên giấy thi của trường Đại Học Bách Khoa Tp.HCM và cuối cùng là tự động nhập điểm số vào danh sách lớp.

# MỤC LỤC

DANH MỤC CÁC HÌNH VẼ .....	VII
DANH MỤC CÁC BẢNG BIỂU .....	IX
DANH MỤC CÁC TỪ VIẾT TẮT.....	X
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI.....	1
1.1 GIỚI THIỆU VỀ ĐỀ TÀI.....	1
1.2 MỤC TIÊU ĐỀ TÀI .....	2
1.3 NỘI DUNG NGHIÊN CỨU.....	3
CHƯƠNG 2. LÝ THUYẾT TỔNG QUAN.....	4
2.1 TỔNG QUAN VỀ HỆ THỐNG NHẬN DẠNG CHỮ VIẾT TAY. ....	4
2.1.1 Các hướng tiếp cận .....	4
2.1.2 Khó khăn và thách thức.....	5
2.1.3 Ứng dụng.....	5
2.1.4 Kiến trúc tổng quát.....	6
2.2 PHÁT HIỆN VĂN BẢN (TEXT DETECTION) .....	6
2.2.1 Tổng quan.....	6
2.2.2 Thuật toán EAST (Efficient and Accurate Scene Text Detection Pipeline) .....	7
2.2.3 Kỹ thuật phân tách từ Scale Space (Scale Space Technique) .....	9
2.3 NHẬN DẠNG VĂN BẢN (TEXT RECOGNITION) .....	10
2.3.1 Tổng quan.....	10
2.3.2 Mô hình Convolutional Recurrent Neural Network (CRNN) kết hợp hàm mất mát Connectionist Temporal Classification (CTC Loss).....	11
2.4 NGÔN NGỮ/PHẦN MỀM/THƯ VIỆN.....	17
2.4.1 Ngôn ngữ Python.....	17
2.4.2 Thư viện.....	17
2.4.3 Google Colab/ Kaggle.....	17
2.4.4 Visual Studio .....	18
CHƯƠNG 3. XÂY DỰNG MÔ HÌNH HỆ THỐNG.....	19
3.1 YÊU CẦU CHỨC NĂNG .....	19

3.2	MÔ HÌNH HOẠT ĐỘNG .....	19
3.3	ẢNH ĐẦU VÀO.....	20
3.4	LƯU ĐỒ GIẢI THUẬT CHO VIỆC HUẤN LUYỆN VÀ NHẬN DẠNG CHỮ VIẾT TAY.....	22
3.5	LƯU ĐỒ GIẢI THUẬT KHI NHẬN DẠNG TRÊN VIDEO .....	24
3.6	GIAO DIỆN NGƯỜI DÙNG .....	25
<b>CHƯƠNG 4. KẾT QUẢ VÀ ĐÁNH GIÁ .....</b>		<b>29</b>
4.1	CƠ SỞ DỮ LIỆU .....	29
4.1.1	Bộ dữ liệu HANDS-VNOnDB2018.....	29
4.1.2	Bộ dữ liệu MNIST.....	29
4.1.3	Bộ dữ liệu thực (tự thu gom) .....	30
4.2	TIỀN XỬ LÝ .....	32
4.3	THUẬT TOÁN EAST VÀ KỸ THUẬT SCALE SPACE .....	36
4.4	NHẬN DẠNG VỚI MÔ HÌNH CRNN KẾT HỢP CTC LOSS .....	38
4.4.1	Thông số cài đặt.....	38
4.4.2	Đánh giá trên tập 122 ảnh .....	40
4.4.3	Đánh giá trên tập 100 ảnh không hạn chế.....	41
4.4.4	Đánh giá trên tập 103 ảnh hạn chế.....	45
4.4.5	Đánh giá trên Video.....	46
<b>CHƯƠNG 5. KẾT LUẬN .....</b>		<b>48</b>
5.1	TÓM TẮT VÀ KẾT LUẬN CHUNG. ....	48
5.1.1	Đóng góp của luận văn .....	48
5.1.2	Hạn chế của luận văn.....	48
5.2	HƯỚNG PHÁT TRIỂN .....	49
<b>TÀI LIỆU THAM KHẢO .....</b>		<b>50</b>

## DANH MỤC CÁC HÌNH VẼ

Hình 1. 1 Giấy thi trường Đại học Bách Khoa và vị trí thông tin.....	2
Hình 2. 1 Các hướng nghiên cứu chính của bài toán nhận dạng chữ.....	4
Hình 2. 2 Mô hình nhận dạng chữ viết tay tổng quát.....	6
Hình 2. 3 Kiến trúc EAST.....	8
Hình 2. 4 Trước (trái) và sau (phải) giai đoạn NMS.....	9
Hình 2. 5 Ảnh các Blob màu trắng.....	10
Hình 2. 6 Ảnh đã được phân tách từ .....	10
Hình 2. 7 Kiến trúc mô hình CRNN .....	11
Hình 2. 8 Kiến trúc CNN .....	12
Hình 2. 9 Mối quan hệ giữa Feature Sequence, Feature Vector và Receptive Field	13
Hình 2. 10 Mạng LSTM (trái) và Bi – LSTM (phải).....	14
Hình 2. 11 Đầu ra của Bi – LSTM theo mỗi Timestep.....	15
Hình 3. 1 Mô hình tổng quát hệ thống nhập điểm tự động từ ảnh bài thi.....	19
Hình 3. 2 Cách lắp đặt Camera .....	20
Hình 3. 3 Hình ảnh minh họa cho ảnh từ ứng dụng IP Webcam .....	21
Hình 3. 4 Lưu đồ giải thuật cho việc huấn luyện và nhận dạng chữ viết tay.....	22
Hình 3. 5 Lưu đồ giải thuật khi nhận dạng trên Video .....	24
Hình 3. 6 Giao diện người dùng.....	25
Hình 3. 7 Hộp thoại hiển thị sau khi bấm nút Browser Files.....	26
Hình 3. 8 Giao diện sau khi nhập đường dẫn.....	27
Hình 3. 9 Giao diện hiển thị trong quá trình làm việc .....	27
Hình 3. 10 Giao diện hiển thị trên Terminal .....	28
Hình 3. 11 Hình ảnh hiển thị kết quả nhận dạng gồm điểm số và MSSV .....	28
Hình 4. 1 Ảnh chữ viết tay trong bộ dữ liệu HANDS – VNOnDB2018 .....	29
Hình 4. 2 Ảnh mẫu MNIST.....	29
Hình 4. 3 Dữ liệu huấn luyện MSSV .....	30
Hình 4. 4 Dữ liệu huấn luyện điểm số .....	30
Hình 4. 5 Ảnh kiểm tra bị hạn chế .....	31
Hình 4. 6 Ảnh kiểm tra không bị hạn chế .....	31
Hình 4. 7 Danh sách lớp File Excel .....	32



Hình 4. 8 Ảnh sau khi dùng kỹ thuật Image Alignment .....	33
Hình 4. 9 Ảnh sau khi xác định khung điểm.....	33
Hình 4. 10 Ảnh các thông tin sau khi được cắt.....	34
Hình 4. 11 Trước và sau khi tăng độ tương phản.....	34
Hình 4. 12 Adaptive Threshold (trái) và OTSU Threshold (phải).....	35
Hình 4. 13 Ảnh họ tên đã được lấy ngưỡng, xóa hàng kẻ và giảm nhiễu.....	35
Hình 4. 14 Ảnh Họ và tên sau khi được phân tách từ.....	35
Hình 4. 15 Ảnh trước và sau khi xóa đường tròn bao quanh.....	35
Hình 4. 16 Đầu ra EAST với ảnh đầu vào có kích thước 640x640 Pixels.....	36
Hình 4. 17 Kết quả EAST 1 .....	36
Hình 4. 18 Đầu ra EAST với ảnh đầu vào có kích thước 1024x640 Pixels.....	37
Hình 4. 19 Kết quả EAST 2 .....	37
Hình 4. 20 Kích thước ảnh, thời gian và kết quả thực hiện bằng EAST .....	37
Hình 4. 21 Kết quả khi thực hiện phân tách từ bằng kỹ thuật Scale Space .....	38
Hình 4. 22 Ảnh chuỗi số huấn luyện.....	38
Hình 4. 23 Model nhận dạng được sử dụng trong đề tài.....	39
Hình 4. 24 Quá trình huấn luyện Model họ và tên.....	40
Hình 4. 25 Quá trình huấn luyện Model MSSV .....	41
Hình 4. 26 Ảnh nhận dạng lỗi vì không có dữ liệu trong Data Train .....	42
Hình 4. 27 Lỗi điểm số vượt ra khỏi vòng tròn giới hạn .....	43
Hình 4. 28 Lỗi cắt sai vị trí ảnh. (Contour được xác định có màu đen đậm).....	43
Hình 4. 29 Lỗi vẽ contour (Contour được xác định có màu đen đậm) .....	44
Hình 4. 30 Lỗi ảnh nhòe gây mất thông tin .....	44
Hình 4. 31 Dấu “,” làm số 0 nhầm thành 9 .....	44
Hình 4. 32 Số 3 nét ngang không có trong tập Train .....	44
Hình 4. 33 Chữ số 5 khó nhận dạng, dấu gạch dưới gây nhiễu .....	44
Hình 4. 34 Số 4 bị nhầm thành số 7 .....	44
Hình 4. 35 Hình ảnh giấy thi mẫu .....	47
Hình 4. 36 Minh họa kết quả nhận dạng trên Terminal .....	47

## **DANH MỤC CÁC BẢNG BIỂU**

Bảng 4. 1 Thông số cài đặt cho Model họ tên và Model số.....	39
Bảng 4. 2 Kết quả nhận dạng họ tên trên tập 122 ảnh .....	40
Bảng 4. 3 Kết quả nhận dạng MSSV và điểm số trên tập 122 ảnh .....	40
Biểu đồ 4. 1 Phân bố điểm số trong danh sách lớp 245 sinh viên .....	32
Biểu đồ 4. 2 Kết quả nhận dạng số thứ tự sinh viên trên tập 100 ảnh không hạn chế .....	42
Biểu đồ 4. 3 Kết quả nhận dạng điểm số của sinh viên trên tập 100 ảnh không hạn chế .....	43
Biểu đồ 4. 4 Kết quả nhận dạng số thứ tự sinh viên trên tập 103 ảnh hạn chế .....	45
Biểu đồ 4. 5 Kết quả nhận dạng điểm số trên tập 103 ảnh hạn chế .....	46

## DANH MỤC CÁC TỪ VIẾT TẮT

AdaBoost	Adaptive Boosting
AI	Artificial Intelligence
CNN	Covolution Neural Network
Concat	Concatenate
CPU	Central Processing Unit
CRNN	Convolutional Recurrent Neural Network
CTC	Connectionist Temporal Classification
DCNN	Deep Convolutional Neural Network
EAST	Efficient and Accurate Scene Text detection pipeline
FCN	Fully – Convolutional Neural Network
FPS	Frames Per Second
GPU	Graphics Processing Unit
JPG	Joint Photographic Experts Group
LSTM	Long Short Term Memory networks
MNIST	Modified National Institute of Standards and Technology
MSSV	Mã số sinh viên
NMS	Non – Max Suppression
OCR	Optical Character Recognition
OpenCV	Open Source Computer Vision Library.
PNG	Portable Network Graphics
RNN	Recurrent Neural Network
SVM	Support Vector Machines
VGG16	Visual Geometry Group from Oxford
YOLO	You Only Look Once

## CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

### 1.1 Giới thiệu về đề tài

Đi cùng với sự phát triển của khoa học công nghệ, tiến lên cách mạng công nghiệp 4.0, nhu cầu tự động hóa để giảm tải công việc cho con người. Hiện nay các ứng dụng từ AI, Machine Learning đã len lỏi vào từng ngõ ngách trong cuộc sống, làm tiện lợi hơn cho những hoạt động thường ngày cũng như trong công việc. Ví dụ, có thể kể đến việc nhận dạng biển số xe bằng máy móc thay vì phải viết giấy như 5 – 10 năm về trước, vừa nhanh chóng, chính xác, lại đảm bảo về mặt an ninh. Các công ty công nghệ lớn giờ đây đã có thể sản xuất ra những chiếc xe chạy tự động, gần như không cần đến sự can thiệp của con người. Hay những hệ thống báo khói, báo cháy thông qua Camera nhận diện hình ảnh giúp báo động và phản ứng nhanh hơn với những trường hợp khẩn cấp...

Tuy nhiên lĩnh vực giáo dục vẫn chưa được đầu tư đúng mức, xứng đáng với những ảnh hưởng mà nó sẽ mang lại. Ngoài những giờ dạy và học trên giảng đường, lớp học. Giáo viên vẫn phải bỏ ra quá nhiều thời gian làm những việc không liên quan đến công việc giảng dạy, truyền đạt kiến thức. Cụ thể là việc nhập điểm số.

Theo thông tư số 58/2011/TT – BGDDT của Bộ Giáo dục và Đào tạo. Chẳng hạn lớp 9 có 7 bài kiểm tra định kỳ (từ 1 tiết trở lên) với 4 bài kiểm tra thường xuyên (15 phút) là 11 bài kiểm tra. Trung bình mỗi kỳ có khoảng gần 500 bài kiểm tra/lớp nếu như mỗi lớp có từ 40 – 45 học sinh. Theo phân công bình thường ở các nhà trường hiện nay thì mỗi giáo viên dạy khoảng 2 lớp 9 và 2 lớp của 3 khối còn lại. Như vậy, mỗi học kỳ thì giáo viên phải chấm khoảng gần 2000 bài kiểm tra.

Với gần 2000 bài kiểm tra, chưa nói đến việc chấm thi. Khối lượng công việc khi nhập điểm vào File Excel cũng đã rất lớn. Mỗi lần chấm bài xong, giáo viên phải rà soát lại bài, tính tổng điểm, vào File Excel để tìm tên, nhập điểm, rồi lại rà soát lại rất mất thời gian, chưa kể những ảnh hưởng đến sức khỏe sau khi làm việc trong thời gian dài... dẫn đến giáo viên có thể nhập điểm sai cho học sinh.

Trên cơ sở đó, ở đề tài “Hệ thống nhập điểm tự động từ ảnh bài thi” em muốn tạo ra một công cụ, có thể phần nào đó giúp đỡ các giáo viên nhập điểm vào File Excel (sau khi giáo viên chấm bài xong) một cách tự động. Đồng thời đề tài này có thể sẽ là nền tảng kiến thức giúp em hoàn thiện được công cụ này trong tương lai.

## 1.2 Mục tiêu đề tài

### ➤ Mục tiêu tổng quát:

Tìm hiểu cách thức vận hành của quá trình nhận dạng chữ viết tay, các mô hình, thuật toán liên quan. Nghiên cứu các vấn đề thực tiễn cần giải quyết, đưa ra phương án và công cụ giải quyết tình trạng quá tải trong việc nhập điểm cho giáo viên.

### ➤ Mục tiêu cụ thể:

Đề tài sẽ thực hiện dò văn bản và nhận diện văn bản từ mẫu giấy thi của trường Đại học Bách Khoa Tp.HCM. Cụ thể, cần nhận dạng được ba thông tin chính “Họ và tên SV”, “MSSV”, “Điểm tổng kết” là các vị trí được khoanh ô màu xanh lam theo hình dưới.

The image shows a sample exam paper from the University of Science, Ho Chi Minh City. The paper contains the following fields and sections:

- Header:** Đại Học Quốc Gia Tp.Hồ Chí Minh, TRƯỜNG ĐẠI HỌC BÁCH KHOA.
- Student Information:**
  - Họ và tên SV: [Green box]
  - MSSV: [Green box]
  - Mi MH: [Green box]
  - Nhóm-Tổ: [Green box]
  - Số sđ: [Green box]
- Exam Subject:** MÔN THI: [Green box]
- Grading Section:**
  - Điểm tổng kết: [Green box]
  - CÁN BỘ CHẤM THI KÝ TÊN: [Green box]
  - CÁN BỘ CỎI THỦ KÝ VÀ GHI RÕ HỌ TÊN: CÁN BỘ CỎI THỦ 1, CÁN BỘ CỎI THỦ 2
- Answer Table:** A table with 10 rows and 2 columns. The first column is labeled 'Câu' (Question) and the second column is for the answer.

Hình 1. 1 Giấy thi trường Đại học Bách Khoa và vị trí thông tin

Giấy thi sau khi được giáo viên chấm xong, sẽ được để ngay dưới góc nhìn của Camera với các điều kiện:

- Ánh sáng bình thường, tránh ngược sáng, ánh sáng điện.
- Góc ảnh: trực diện hoặc góc nghiêng không quá 10 độ.
- Không bị che khuất

Sau khi nhận dạng, đối chiếu với thông tin MSSV và họ tên sinh viên có sẵn trong danh sách lớp để tự động nhập điểm vào File Excel.

Hiện thị giao diện để thầy/cô có thể làm việc, tương tác với hệ thống.

### **1.3 Nội dung nghiên cứu**

Đề tài “Hệ thống nhập điểm tự động từ ảnh bài thi” gồm các nội dung cụ thể như sau:

- Tìm hiểu tổng quan về hệ thống nhận dạng chữ viết tay và các thành phần chính.
- Xây dựng mô hình giải thuật phát hiện chữ viết, nhận dạng chữ viết tay.
- Xây dựng phần mềm tự động nhập điểm vào danh sách lớp File Excel.
- Xây dựng giao diện người dùng.
- Hiện thực hóa bằng mô hình trực quan.

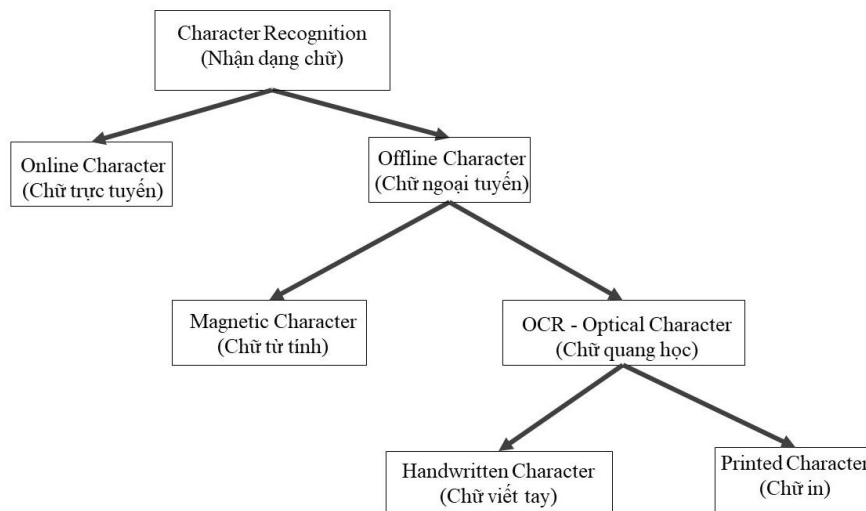
## CHƯƠNG 2. LÝ THUYẾT TỔNG QUAN

### 2.1 Tổng quan về hệ thống nhận dạng chữ viết tay.

#### 2.1.1 Các hướng tiếp cận

Nhận dạng chữ là lĩnh vực được nhiều nhà nghiên cứu quan tâm và cho đến nay lĩnh vực này cũng đã đạt được nhiều thành tựu lớn lao cả về mặt lý thuyết lẫn ứng dụng thực tế. Lĩnh vực nhận dạng chữ được chia làm hai nhánh chính: nhận dạng chữ trực tuyến (Online Character Recognition) và nhận dạng chữ ngoại tuyến (Offline Character Recognition)

Do dữ liệu đầu vào là ảnh văn bản nên nhận dạng chữ viết tay ngoại tuyến và nhận dạng chữ in còn được gọi chung là nhận dạng chữ quang học OCR (Optical Character Recognition).



Hình 2. 1 Các hướng nghiên cứu chính của bài toán nhận dạng chữ

Nhận dạng chữ viết tay trực tuyến được thực hiện bằng cách lưu lại các thông tin về nét chữ như thứ tự, nét viết, hướng và tốc độ của nét viết trong quá trình viết. Trong khi đó nhận dạng chữ viết tay ngoại tuyến có dữ liệu đầu vào là ảnh văn bản được quét vào nên việc nhận dạng có độ khó cao hơn nhiều so với nhận dạng chữ viết tay trực tuyến.

Đến thời điểm này, nhận dạng chữ in và chữ viết tay trực tuyến đã được giải quyết gần như trọn vẹn. Tuy nhiên, bài toán nhận dạng chữ viết tay ngoại tuyến vẫn đang là thách thức lớn đối với các nhà nghiên cứu.

### 2.1.2 Khó khăn và thách thức

Khó khăn lớn nhất khi nghiên cứu bài toán nhận dạng chữ viết tay nói chung hay chữ viết tay ngoại tuyến nói riêng là chữ viết của mỗi người một khác, có những trường hợp đến cả người cũng không thể đọc được rõ ràng. Các chữ bị xiên, vẹo, dính liền nhau. Chữ viết bị sai chính tả, kiểu viết thay đổi theo thói quen, thời gian nên khó khăn trong việc trích chọn đặc trưng.

Tiếng Việt cũng có những khó khăn riêng của chính nó. Tuy cùng sử dụng bộ chữ cái gần giống với bộ chữ La Tinh đã khá là phổ biến trên các diễn đàn học thuật. Tiếng Việt còn có các chữ cái khác như ă, â, ô, ơ, ư, ê và tập các dấu câu huyền, sắc, hỏi, ngã, nặng. Nếu nhận dạng không tốt, máy sẽ bị nhầm lẫn về mặt ngữ nghĩa. Đồng thời, có khá ít bộ dữ liệu để huấn luyện các mô hình nhận dạng chữ viết tay tiếng Việt.

### 2.1.3 Ứng dụng

Cho đến nay, bài toán nhận dạng chữ viết tay đã được áp dụng trong nhiều ứng dụng thực tế. Nhận dạng chữ viết tay đã được áp dụng trong lĩnh vực an ninh, hành chính công, dịch thuật... với một số ứng dụng điển hình bao gồm:

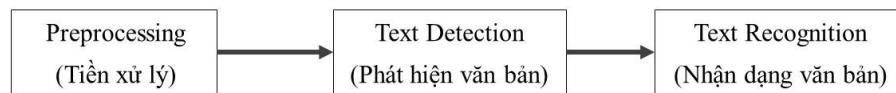
- Chuyển đổi văn bản viết tay thành văn bản kỹ thuật số
- Xác thực chữ kí
- Dịch thuật tự động
- Trích xuất trường thông tin cần lấy từ văn bản hành chính như tờ trình, công văn, quyết định

Như vậy có thể thấy rằng nhận dạng chữ viết tay có ứng dụng quan trọng và có tiềm năng lớn để áp dụng rộng rãi trong nhiều tác vụ trong cuộc sống.



### 2.1.4 Kiến trúc tổng quát

Một hệ thống nhận dạng chữ viết tay thường gồm các phần sau:



**Hình 2. 2 Mô hình nhận dạng chữ viết tay tổng quát**

Giai đoạn tiền xử lý thường bao gồm các bước loại bỏ nhiễu khỏi hình ảnh, xóa nền phức tạp khỏi hình ảnh, xử lý các điều kiện khác nhau trong ảnh. Xoay ảnh về đúng góc độ để hỗ trợ các giai đoạn sau.

Giai đoạn dò/phát hiện văn bản giúp khoanh vùng, xác định vị trí có văn bản, hỗ trợ cho việc tách văn bản ra khỏi nền (những thông tin dư thừa) hỗ trợ cho việc nhận dạng văn bản tốt hơn. Thông thường giai đoạn này đã bao gồm các bước phân tách hàng (Line Segmentation) và phân tách từ (Word Segmentation).

Nhận dạng văn bản giúp chuyển đổi thông tin hình ảnh sang dạng ASCII để làm việc, hỗ trợ cho các ứng dụng/phần mềm theo sau. Giai đoạn này có thể bị ảnh hưởng rất lớn bởi quá trình phân đoạn kí tự trong một từ (Character Segmentation).

## 2.2 Phát hiện văn bản (Text Detection)

### 2.2.1 Tổng quan

Nhiệm vụ của phát hiện văn bản là vẽ được các Bounding Box (hình chữ nhật hoặc hình tứ giác) bao quanh các dòng chữ có trong ảnh. Một số khó khăn và thách thức của việc phát hiện văn bản cũng như văn bản viết tay trong đề tài này như sau:

- Sự đa dạng về kiểu kí tự: kí tự có kiểu dáng, kích cỡ, màu mực, độ nghiêng khác nhau.
- Nền ảnh khác nhau: Giấy thi có nền trắng với nhiều cấp độ khác nhau, hoặc các dòng kẻ làm nhiễu thông tin
- Một số nhân tố khác: Ảnh bị mờ, kém chất lượng, các loại nhiễu như nhiễu hạt, nhiễu muối tiêu...

Có 3 kỹ thuật chính để phát hiện văn bản:

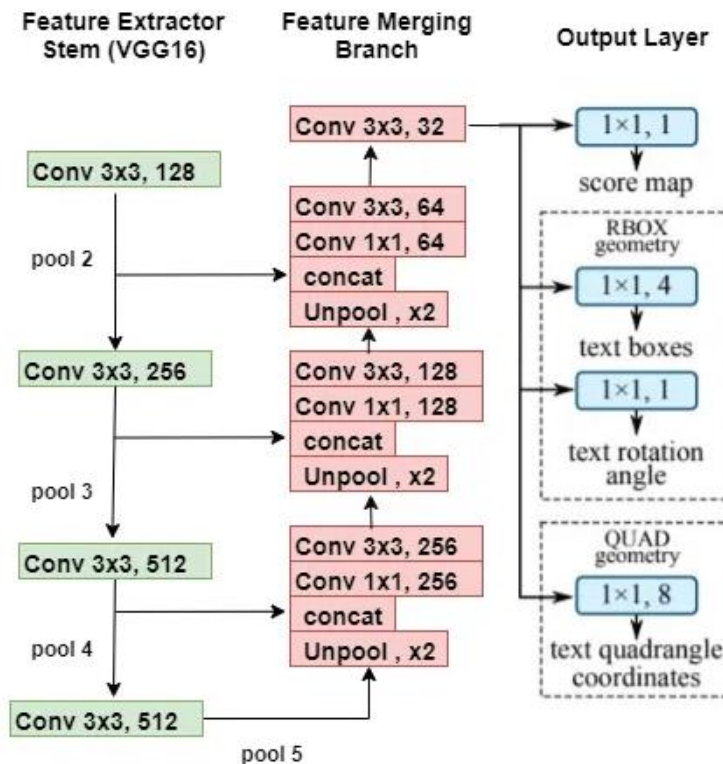
- Dựa vào Texture (Texture Based Method): Coi văn bản như một thông tin có cấu trúc và có thuộc tính riêng khác với hình ảnh và nền. Phù hợp cho các văn bản nằm ngang và có thể giải quyết tốt vấn đề tỉ lệ (Scale). Ít bị ảnh hưởng bởi nhiễu, màu sắc, nền. Tuy nhiên, có nhược điểm là thời gian tính toán dài.
- Dựa vào Component (Component Based Method): Phân chia ảnh thành các cụm thông tin dựa vào màu sắc, biên để chọn vùng có khả năng chứa văn bản... sau đó lọc bớt nhờ ngưỡng hoặc các bộ phân loại. Phương pháp này gặp khó khăn với nhiễu, văn bản có nhiều màu sắc khác nhau, văn bản được viết trên mặt phẳng gồ ghề, nhưng bù lại có khả năng xử lý nhanh chóng và khả năng giải quyết tốt vấn đề về tỉ lệ và phông chữ
- Sử dụng kết hợp 2 kỹ thuật trên (Hybrid Method): Sử dụng các thuộc tính Gradient và Geometrical để xác định vùng có chứa văn bản.

Rất nhiều phương pháp đã được đề xuất và chứng minh là có hiệu quả trong việc phát hiện văn bản với độ chính xác cao lên đến hơn 90% như Adaboost, SVM, CNN, YOLO... Tuy nhiên các kiến trúc được đưa ra rất phức tạp, gồm nhiều tầng, lớp khiến cho việc xử lý bị chậm khó có thể áp dụng vào các công nghệ thời gian thực.

### **2.2.2 Thuật toán EAST (Efficient and Accurate Scene Text Detection Pipeline)**

Thuật toán bao gồm 2 giai đoạn chính. Giai đoạn một là một mạng FCN (Fully – Convolutional Neural Network) để trực tiếp nhận biết vùng chứa văn bản, cho ra kết quả là xác suất chứa văn bản và vị trí của Text Box. Tiếp theo là giai đoạn NMS (Non – Max Suppression) để gộp các Text Box thành một Bounding Box quanh văn bản.

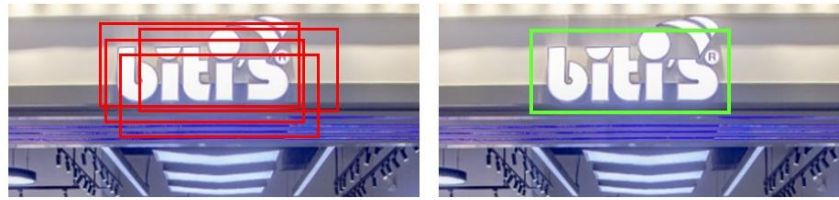
Mạng FCN gồm 3 phần chính: Lớp trích tách đặc trưng (Feature Extractor), lớp ghép các đặc trưng (Feature Merging) và lớp đầu ra (Output Layer).



Hình 2. 3 Kiến trúc EAST

- Lớp tách đặc trưng: có thể sử dụng các mô hình mạng tích chập (Convolutional Network) được huấn luyện sẵn, mô hình PVANet hoặc mô hình VGG16. Cho ra các Feature Map với kích thước nhỏ hơn ảnh đầu vào.
- Lớp ghép các đặc trưng: các Feature Map từ lớp trước được đưa vào lớp Unpooling để tăng gấp đôi kích thước. Sau đó được đưa vào lớp Concat (Concatenate) để ghép với Feature Map hiện có. Lớp Conv 1x1 nhằm giảm bớt chi phí tính toán, lớp Conv 3x3 để làm ngưỡng loại bỏ các Feature Map không đủ yêu cầu.
- Lớp đầu ra nhận Feature Map của lớp trước. Thông qua nhiều lớp Conv 1x1 sẽ biến đổi Feature Map 32 kênh thành một kênh Score Map mang thông tin xác suất chứa văn bản và các kênh Geometry Map chứa vị trí và góc quay của văn bản.

Giai đoạn NMS: Ghép lần lượt các lớp Geometry Map lại với nhau theo từng hàng vì thường các Pixel nằm cạnh nhau theo hàng ngang có sự kết nối lớn hơn.



Hình 2. 4 Trước (trái) và sau (phải) giai đoạn NMS

➤ **Ưu điểm:**

Nhờ vào việc loại bỏ bớt các lớp, giai đoạn trung gian không cần thiết mà thuật toán EAST đã trở thành một trong những thuật toán dò văn bản vượt trội cả về độ chính xác và tốc độ xử lý. EAST có thể phát hiện văn bản cả trong các ứng dụng thời gian thực ở mức 13 FPS trên hình ảnh 720p với độ chính xác cao.

➤ **Khuyết điểm:**

Thuật toán EAST chỉ phù hợp với văn bản phi cấu trúc. Ở đề tài này, các thông tin trên giấy thi luôn nằm tại vị trí xác định, hoàn toàn có thể được cắt thủ công và sau đó áp dụng các phương pháp phân tách từ (Word Segmentation) với chi phí tính toán ít hơn và thời gian xử lý ngắn hơn.

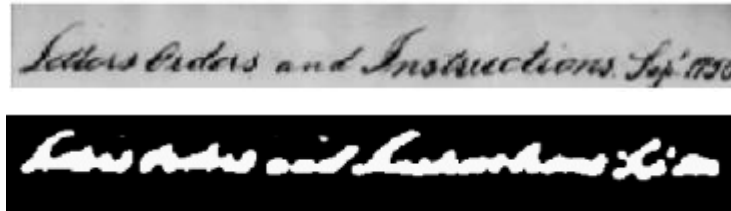
### 2.2.3 Kỹ thuật phân tách từ Scale Space (Scale Space Technique)

Kỹ thuật này được đưa ra bởi Manmatha and Srimal lần đầu vào năm 1999 nhưng vẫn cho ra kết quả nhanh chóng, chính xác và phù hợp với mục tiêu phát hiện, phân tách chữ viết trong đề tài này.

Với một bức ảnh đầu vào là một đoạn văn viết tay với ảnh đầu ra kèm theo các từ đã được thì kỹ thuật sẽ đi qua ba bước chính: Tiền xử lý, phân tách dòng (Line Segmentation) và phân tách từ (Word Segmentation).

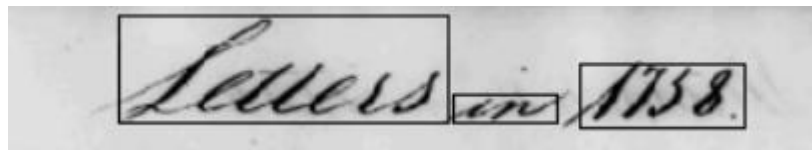
- Tiền xử lý: Chuyển về ảnh xám, lọc nhiễu và đặc biệt là xóa đi các dòng kẻ ô ngang, dọc (nếu có).
- Phân tách dòng (Line Segmentation): Sử dụng phương pháp tách dùng lược đồ sáng (Histogram) với ý tưởng là ở phần có mực viết đi qua thì khu vực đó sẽ tối hơn.

- Phân tách từ (Word Segmentation): Trong một từ có những chữ cái được viết liền hoặc viết tách. Sau đó sử dụng các bộ lọc Gauss theo cả trục ngang và dọc để làm giãn nở các chữ cái tạo thành các Blob. Khi đó từng từ sẽ trông như các đốm màu trắng.



Hình 2. 5 Ảnh các Blob màu trắng

Cuối cùng là vẽ đường bao quanh các Blob này và trích xuất ảnh của từng từ.



Hình 2. 6 Ảnh đã được phân tách từ

➤ **Ưu điểm:**

Phương pháp này nhanh, phù hợp với các văn bản có cấu trúc. Đồng thời có chi phí tính toán thấp, phù hợp với nhiệm vụ mà đề tài luận văn đề ra.

➤ **Khuyết điểm:**

Việc chọn thủ công các thông số, tỉ lệ phù hợp để các từ trong cùng một hàng không dính liền với nhau khá là phức tạp, yêu cầu nhiều thời gian thử nghiệm.

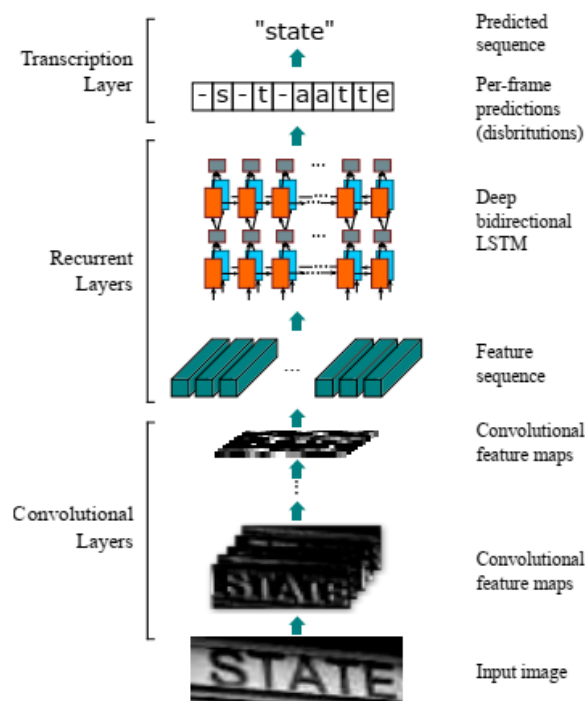
## 2.3 Nhận dạng văn bản (Text recognition)

### 2.3.1 Tổng quan

Các phương pháp nhận dạng văn bản bao gồm các phương pháp như đối sánh mẫu, mạng thần kinh, mô hình Markov ẩn, máy véc tơ lựa... Tuy nhiên, những phương pháp này không còn cho độ chính xác cao. Gần đây người ta sử dụng các kiến trúc lai ghép hoặc song song để đạt được độ chính xác cao với thời gian xử lý ngắn hơn bằng cách sử dụng mô hình CNN kết hợp SVM, mô hình CRNN kết hợp CTC Loss hay sử dụng Tesseract OCR...

### 2.3.2 Mô hình Convolutional Recurrent Neural Network (CRNN) kết hợp hàm mất mát Connectionist Temporal Classification (CTC Loss)

Mạng thần kinh tái phát liên tục (CRNN) là sự kết hợp của mạng CNN, RNN và CTC Loss cho các tác vụ nhận dạng chuỗi với độ dài bất kì như nhận dạng văn bản, phân loại Video, phân loại hành động...



Hình 2. 7 Kiến trúc mô hình CRNN

Kiến trúc mô hình gồm 3 giai đoạn chính bao gồm mạng thần kinh tích chập (Convolution Neural Network), mạng hai chiều LSTM (Bidirectional Long Short Term Memory Network), lớp phiên mã (Transcription Layer).

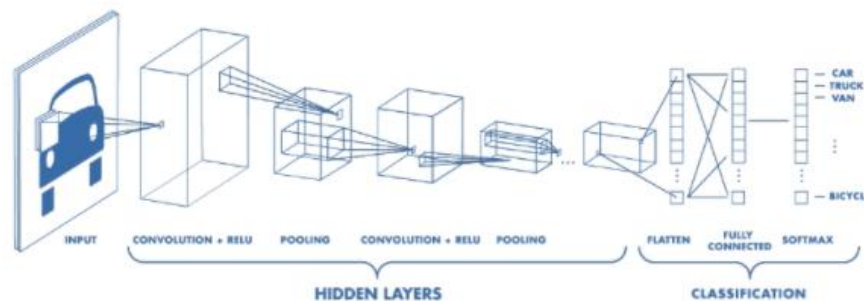
#### 2.3.2.1 Mạng thần kinh tích chập (Convolutional Neural Network)

CNN bao gồm tập hợp các lớp cơ bản bao gồm: Lớp tích chập (Convolution Layer), Lớp phi tuyến (Nonlinear Layer), Pooling Layer, Fully Connected Layer. Các lớp này liên kết với nhau theo một thứ tự nhất định.

- Lớp tích chập (Convolution Layer) là lớp quan trọng nhất và cũng là lớp đầu tiên của của mô hình CNN. Lớp này có chức năng chính là phát hiện các đặc

trung có tính không gian hiệu quả gồm góc, cạnh, màu sắc, hoặc Texture của ảnh mà không bị ảnh hưởng bởi các phép tỉ lệ, xoay...

- Lớp phi tuyến (Nonlinear Layer) giúp hạn chế tình trạng Vanishing Gradient, phát hiện ra những quan hệ phức tạp của dữ liệu. Có nhiều hàm kích hoạt phi tuyến như Tanh, Sigmoid nhưng phổ biến nhất bây giờ là hàm ReLU.
- Lớp Pooling thường được dùng giữa các lớp tích chập, để giảm kích thước dữ liệu nhưng vẫn giữ được các thuộc tính quan trọng. Đồng thời giảm công việc tính toán trong mô hình.
- Ở lớp Fully Connected, dữ liệu đầu ra của lớp đứng trước sẽ được là phẳng thành Vector và đưa vào một lớp được kết nối như một mạng thần kinh. Cuối cùng sử dụng hàm Softmax hoặc Sigmoid dùng để phân loại đầu ra.

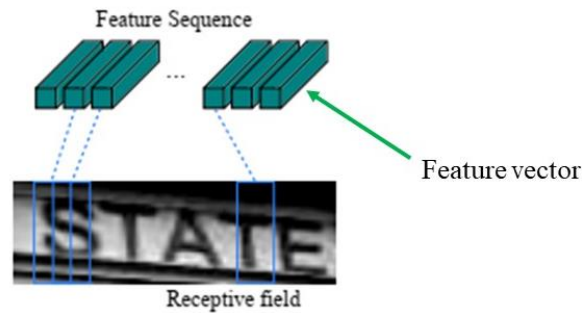


Hình 2. 8 Kiến trúc CNN

Mạng CRNN được xây dựng bằng cách giữ lại lớp tích chập (Convolutional Layer) và lớp Max Pooling từ mô hình CNN (bỏ đi lớp Fully Connected Layers) nhằm trích xuất các đặc trưng từ hình ảnh đầu vào.

Từ ảnh gốc khi qua các lớp tích chập sẽ tạo ra được các Feature Map và từ đó một chuỗi các Feature Vector sẽ được tạo ra gọi là Feature Sequence để đưa vào mạng RNN. Mỗi Feature Vector có thể tương ứng với một vùng Receptive Field ở ảnh gốc.

Mọi hình ảnh phải được chuẩn hóa về cùng chiều cao với độ dài có thể khác nhau. Như ở mô hình CNN thuần túy, hình ảnh thường được chuẩn hóa về cùng kích cỡ cả về chiều cao lẫn chiều dài để có cùng số chiều đầu vào. Thế nên không phù hợp với thông tin dạng chuỗi khi mà độ dài của mỗi chuỗi văn bản là khác nhau.



Hình 2. 9 Mối quan hệ giữa Feature Sequence, Feature Vector và Receptive Field

### 2.3.2.2 Mạng hai chiều LSTM (Bidirectional Long Short Term Memory Network)

Lớp này dự đoán các kí tự từ mỗi Feature Vector và một số mối quan hệ giữa các kí tự có ở lớp trước tích chập trước.

Mạng RNN có 3 lợi ích sau:

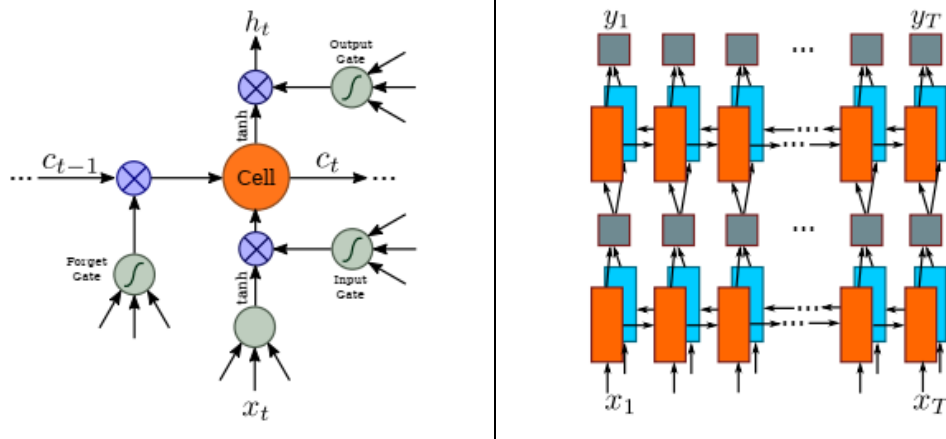
- Sử dụng ngữ cảnh để nhận dạng chuỗi (sửa và cập nhật liên tục) thay vì nhận dạng từng kí tự một cách độc lập.
- Có thể dùng cơ chế truyền ngược (Back Propagation) cho phép huấn luyện cả mạng RNN và CNN cùng một lúc.
- Có thể hoạt động trên các chuỗi có độ dài tùy ý.

Tuy nhiên RNN sẽ gặp vấn đề khi ghi nhớ các kí tự ở quá xa vì mất mát đạo hàm (Vanishing Gradient). Thế nên sẽ cần phải sử dụng mạng LSTM (Long Short Term Memory Network) là một biến thể khác của RNN.

Thành phần chính của LSTM là Cell State, đường nằm ngang  $C_{t-1}$  đến  $C_t$  nó như một dạng băng truyền. Có thể cho phép thông tin đi thẳng từ đầu đến cuối mạng. Đồng thời LSTM có khả năng bỏ bớt hoặc thêm các thông tin vào Cell State thông qua cấu trúc cổng.

Mạng LSTM là một mạng có định hướng (Directional Network), tức là nó chỉ sử dụng thông tin từ quá khứ (kí tự đứng trước). Nhưng với chuỗi văn bản, cả kí tự đứng trước và sau đều có giá trị về mặt nội dung, vì vậy người ta sử dụng kết hợp hai mạng LSTM (truyền tới và truyền ngược) được gọi là mạng hai chiều LSTM (Bi – LSTM). Một số mô hình còn kết hợp nhiều mạng Bi – LSTM để nâng cao độ chính xác.





Hình 2. 10 Mạng LSTM (trái) và Bi – LSTM (phải)

### 2.3.2.3 Lớp phiên mã (Transcription Layer)

Lớp phiên mã giúp chuyển đổi mỗi khung hình (Per – Frame Prediction) được tạo bởi mạng Bi – LSTM ở trước thành chuỗi kết quả cuối cùng. Có hai chế độ phiên mã, đó là phiên mã không có từ vựng (Lexicon Free) là dự đoán chuỗi thuần túy dựa vào xác suất cao nhất có được và phiên mã từ vựng dựa trên từ điển (Lexicon Based) là dự đoán chuỗi dựa vào một từ điển có sẵn.

### 2.3.2.4 Hàm mất mát Connectionist Temporal Classification (CTC Loss)

Ở đầu ra của lớp Bi – LSTM, ta có được xác suất xuất hiện của các kí tự ứng với mỗi Timestep được biểu diễn dưới dạng một ma trận. Có hai việc cần làm với ma trận này, đó là:

- Tính toán hàm mất mát (Loss Function) để huấn luyện mạng CRNN.
- Giải mã ma trận để được chuỗi văn bản đầu ra.

Việc sử dụng CTC Loss sẽ giúp hoàn thành hai nhiệm vụ trên cùng với ưu điểm:

- Chỉ cần có bộ dữ liệu là ảnh kèm nhãn là từ (Word), chứ không cần là các kí tự kèm vị trí của chúng trong ảnh.
- Vì đầu ra LSTM chỉ là xác suất của các kí tự, không hợp lý khi cứ mặc định các kí tự liên nhau sẽ được gộp thành một. Nếu lấy ví dụ là từ ‘to’ hoặc ‘too’ sẽ cho ra kết quả sai lệch.

### 1) Mã hóa nhãn dữ liệu huấn luyện (Encoding Ground Truth)

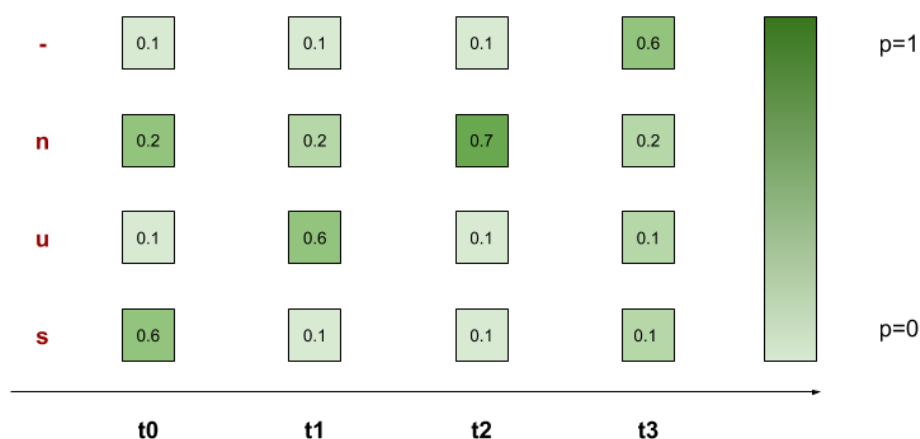
CTC Loss giải quyết vấn đề này theo cách rất thông minh. Cụ thể, nó thử tất cả các Alignment của Ground Truth và tính xác suất của tổng tất cả các Alignment đó. Mạng CRNN sẽ học từ CTC Loss để tự đưa ra những Alignment này bằng cách thêm kí tự trống “-” và lặp lại bất kì kí tự nào trong Ground Truth.

Giả sử Ground Truth là từ sun và có mô hình LSTM dự toán 4 Timesteps. Thì những Alignment đúng của Ground Truth sẽ là: -sun, s-un, su-n, sun-, suun, ssun, sunn. Nhưng không thể là: s-n- (thiếu kí tự u), usn (chỉ có 3 trên 4 kí tự), us-n (không giải mã ra được từ sun).

Đối với những từ có 2 kí tự liên tục giống nhau, sẽ thêm kí tự “-” ở giữa để tạo ra một Alignment đúng. Ví dụ với từ odd, các Alignment đúng có thể là: -od-d, ood-d. Nhưng không thể là ooddd.

### 2) Tính CTC Loss

Với mỗi Ground Truth sẽ có nhiều Alignment, bất kì Alignment nào được dự đoán cũng đều là một dự đoán đúng. Do đó, hàm Loss cần tối ưu tổng của tất cả các Alignment.



Hình 2. 11 Đầu ra của Bi – LSTM theo mỗi Timestep

Với từ sun, có tổng 7 Alignment đúng ở trên. Do đó theo mô hình, xác suất từ sun xuất hiện là:

$$\begin{aligned}
 p('sun') &= p('s-un') + p('s-un') + p('su-n') + p('sun-') + p('ssun') + p('suun') + p('sunn') \\
 &= (0.1 \times 0.1 \times 0.1 \times 0.2) + (0.6 \times 0.1 \times 0.1 \times 0.2) + (0.6 \times 0.6 \times 0.1 \times 0.2) + \\
 &\quad (0.6 \times 0.6 \times 0.7 \times 0.6) + (0.6 \times 0.1 \times 0.1 \times 0.2) + (0.6 \times 0.6 \times 0.1 \times 0.2) + \\
 &\quad (0.6 \times 0.6 \times 0.7 \times 0.2) \\
 &= 0.2186
 \end{aligned}$$

Hàm Loss sẽ là  $1 - p('sun') = 1 - 0.2186 = 0.7814$

Mục tiêu của việc huấn luyện là điều chỉnh các thông số trong mạng CRNN, điều chỉnh lại xác suất đầu ra của từng kí tự theo Timestep sao cho  $p('sun')$  tiệm cận 1. Và hàm Loss về gần bằng 0.

### 3) Giải mã văn bản (Decoding Text)

Có nhiều kỹ thuật để giải mã văn bản như Best Path, Beam Search, Word Beam Search... nhưng chúng đều có chung ý tưởng như sau. Sau khi có được xác suất xuất hiện của các kí tự ứng với mỗi Timestep được biểu diễn dưới dạng một ma trận ở đầu ra của lớp Bi – LSTM. Mô hình sẽ giải mã để đưa ra chuỗi dự đoán cuối cùng bằng cách gộp những kí tự lặp lại liên tiếp nhau thành một kí tự và sau đó xóa hết tất cả kí tự “-“. Ví dụ với Alignment ood-d thì sau khi giải mã sẽ được từ odd bằng cách gộp 2 kí tự ‘o’ lại với nhau và xóa dấu “-“.

#### 2.3.2.5 Ưu và nhược điểm

##### ➤ Ưu điểm:

Có thể học trực tiếp từ ảnh của các từ (Word). Không nhất thiết phải thêm các bước tiền xử lý như nhị phân hóa ảnh (Binarization), phân đoạn kí tự (Character Segmentation), Component Localization... Áp dụng được với chuỗi văn bản có độ dài bất kì. Mang lại hiệu suất nhận dạng cao hơn với thông tin chuỗi. Ít thông số tính toán hơn so với mạng thần kinh tích chập sâu (Deep Convolutional Neural Network).

##### ➤ Nhược điểm:

Không hoạt động tốt đối với các chuỗi ngẫu nhiên, chẳng hạn như “1252\_5”. Không có lịch sử ngữ cảnh cho “mô hình ngôn ngữ” trong trường hợp này.

## 2.4 Ngôn ngữ/phần mềm/thư viện

### 2.4.1 Ngôn ngữ Python

Python được thiết kế với điểm mạnh là dễ đọc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người lập trình. Việc lập trình bằng Python mang lại nhiều lợi thế như:

- Tính đơn giản và nhất quán
- Tính linh hoạt
- Cộng đồng rộng lớn.

Python còn có một kho công nghệ phong phú bao gồm rất nhiều thư viện cho lĩnh vực trí tuệ nhân tạo.

### 2.4.2 Thư viện

Ở Project này em đã sử dụng các thư viện sau:

- Keras, TensorFlow, and Scikit – Learn cho Machine Learning.
- NumPy cho phân tích dữ liệu và tính toán khoa học hiệu năng cao
- Pandas cho phép làm việc dễ dàng với các File csv và Excel
- Matplotlib cho phép làm việc với đồ thị, hình ảnh có tọa độ.
- OpenCV cho xử lý ảnh

Và một số thư viện khác phục vụ cho việc nhập xuất thư mục, in thời gian...

### 2.4.3 Google Colab/ Kaggle

Để viết một chương trình sử dụng Framework về AI/Deep Learning như TensorFlow, Keras. Có thể sử dụng bất kì Python IDE nào như PyCharm, Jupyter Notebook hay Visual Studio để lập trình. Tuy nhiên, việc huấn luyện mô hình yêu cầu hệ thống phải có tốc độ xử lý cao (thông thường dựa trên GPU).

Google Colab hay Kaggle cho phép chạy các dòng Code Python thông qua trình duyệt Web, đặc biệt phù hợp với Data Analysis, Machine Learning và giáo dục. Các chương

trình này không yêu cầu cài đặt hay cấu hình máy tính mà vẫn có thể sử dụng tài nguyên máy tính từ CPU tốc độ cao cho đến GPUs và cả TPUs.

Mặc dù có nhiều ưu điểm như trên nhưng bù lại, việc bị giới hạn thời gian huấn luyện trong ngày bắt buộc người dùng phải thêm thắt các thao tác xử lý phức tạp khác. Em đã sử dụng Kaggle bên cạnh Colab. Kaggle có những ưu điểm như của Colab và thời gian huấn luyện lên đến 40 giờ/tuần. Đồng thời cộng đồng tham gia Kaggle ngày càng tăng cũng là một ưu điểm lớn.

#### **2.4.4 Visual Studio**

Visual Studio là một trong những công cụ hỗ trợ lập trình nổi tiếng nhất hiện nay của Microsoft và chưa có một phần mềm nào có thể thay thế được

Sở dĩ Visual Studio được giới lập trình ưa chuộng như vậy là bởi những ưu điểm vượt trội như sau:

- Visual Studio hỗ trợ lập trình trên nhiều nền tảng ngôn ngữ khác nhau từ C/C++, C#, cho đến F#, Visual Basic, HTML, CSS, JavaScript. Thậm chí, phiên bản VS 2015 có hỗ trợ Code trên ngôn ngữ Python.
- Visual Studio giúp hỗ trợ khả năng gỡ rối (Debug) hiệu quả và dễ dàng
- Visual Studio sở hữu giao diện thân thiện, dễ dàng sử dụng
- Visual Studio cho tích hợp nhiều ứng dụng khác, cho phép cài đặt thư viện dễ dàng nhờ Nuget.

## CHƯƠNG 3. XÂY DỰNG MÔ HÌNH HỆ THỐNG

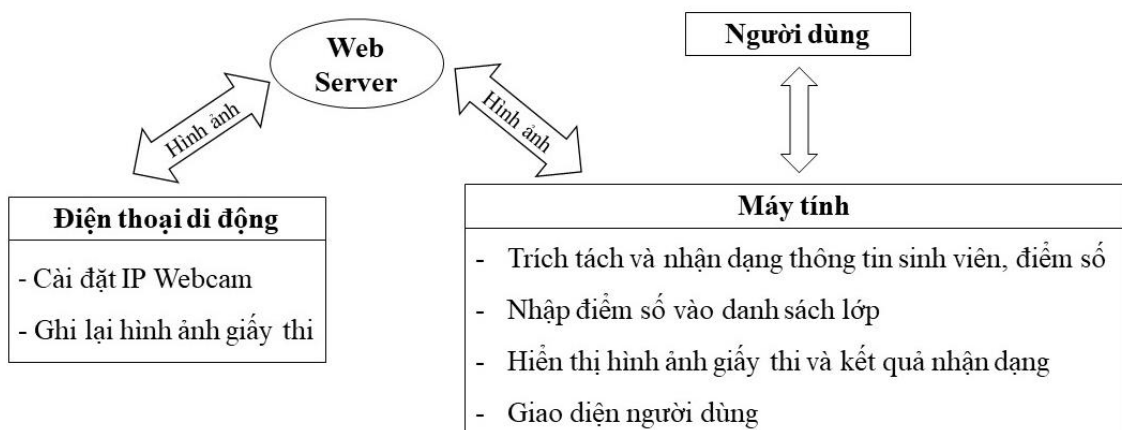
### 3.1 Yêu cầu chức năng

Hệ thống được xây dựng có những chức năng chính như sau:

- Chức năng chụp ảnh và truyền dữ liệu: Sau khi người dùng cài đặt các thông số xử lý, sắp xếp vị trí Camera và nơi đặt để giấy thi. Hình ảnh giấy thi sẽ được Camera chụp lại và truyền liên tục đến máy tính để xử lý thông qua Web Server.
- Chức năng nhận dạng chữ viết tay: Máy tính sau khi nhận được dữ liệu hình ảnh từ Web Server sẽ tiến hành trích tách và nhận dạng các thông tin “Họ và Tên”, “MSSV”, “Điểm tổng kết” để phục vụ cho việc nhập điểm tự động.
- Phần mềm quản lý: Thông tin sau khi nhận dạng sẽ được tự động ghi lại vào danh sách lớp. Sẽ có giao diện giao tiếp giữa người dùng và hệ thống để người dùng có thể nhập địa chỉ của Web Server chứa Video quay lại hình ảnh giấy thi, cũng như là đường dẫn đến danh sách lớp đang có trong máy tính. Thêm vào đó, phần mềm sẽ hiển thị trực tiếp hình ảnh giấy thi kèm kết quả nhận dạng để người sử dụng tiện theo dõi.

### 3.2 Mô hình hoạt động

Mô hình hoạt động của hệ thống nhập điểm tự động từ ảnh bài thi có dạng như sau:



Hình 3. 1 Mô hình tổng quát hệ thống nhập điểm tự động từ ảnh bài thi

Hệ thống gồm có 2 thành phần chính là Camera từ điện thoại và hệ thống nhận dạng là máy tính. Ở đây em sử dụng Camera của điện thoại di động đã cài đặt ứng dụng IP Webcam để ghi lại hình ảnh giấy thi và truyền liên tục thông qua đường truyền Wifi lên Web Server tại địa chỉ cố định.

Máy tính lúc này sẽ liên tục gửi yêu cầu đến Web Server để lấy hình ảnh giấy thi về để xử lý, nhận dạng và nhập điểm vào danh sách lớp. Người dùng sẽ cần phải cài đặt các thông số cho Camera, cài đặt địa chỉ Web Server trên ứng dụng IP Webcam trước và tiếp đó nhập địa chỉ Web Server và đường dẫn đến danh sách lớp ở giao diện làm việc trên máy tính.

Người dùng có thể vừa chấm bài trong khi hệ thống thực hiện việc nhận dạng. Cứ mỗi bài thi được chấm xong, người dùng đặt bài thi đó dưới Camera, thời gian trung bình giữa những lần chấm bài như vậy khoảng 2 phút.

### **3.3 Ảnh đầu vào**

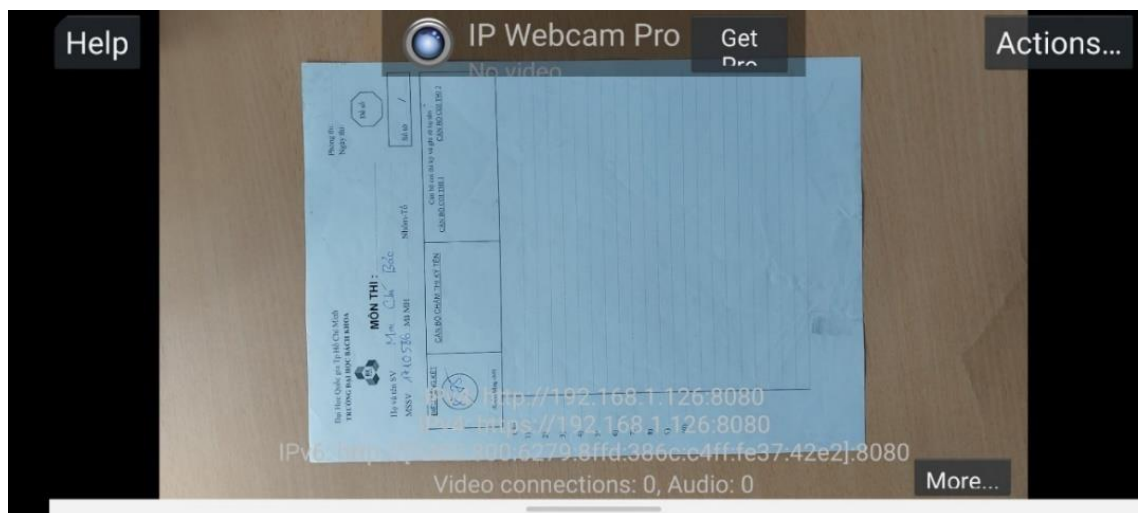
Ảnh đầu vào được thu liên tục bằng Camera điện thoại hướng về khu vực mà giáo viên để giấy thi, ảnh phải có chất lượng cao để giảm thiểu tối đa thông tin nhiễu. Camera được lắp đặt tại vị trí chính diện nhất có thể. Hạn chế những khu vực mà việc ánh sáng quá mạnh hoặc quá yếu có thể làm chói, nhòe, đổ bóng hay mặt bàn gồ ghề, lồi lõm có thể gây ảnh hưởng đến quá trình xác định vị trí thông tin và nhận dạng.



**Hình 3. 2 Cách lắp đặt Camera**

Trong đề tài này em sử dụng điện thoại Samsung Galaxy A52s 5G. Có cài đặt ứng dụng IP Webcam là một phần mềm giúp biến Camera điện thoại thành một IP Camera. Các thông số được cài đặt cho Video từ điện thoại như sau:

- Frames per Second: 20 fps
- Video Resolution: 1920x1080 Pixels
- Khoảng cách từ Camera đến mặt bàn: 15 – 20 cm
- Địa chỉ URL: <http://192.168.1.126:8080/video>
- Flash Mode: Default (chỉ cài đặt chế độ Always use Flash khi không đủ ánh sáng)

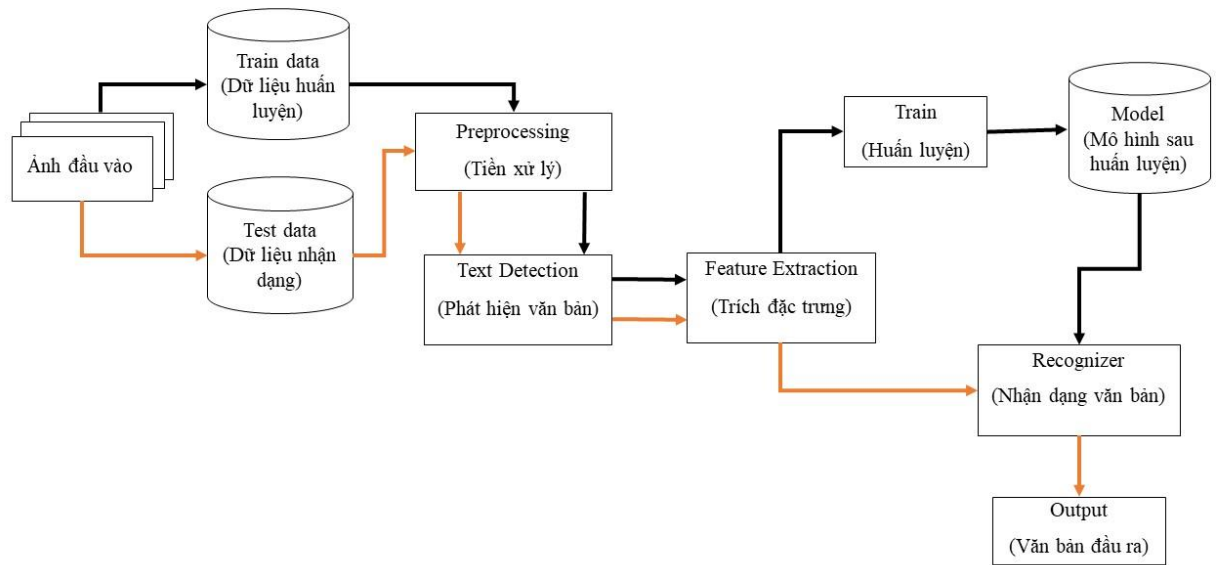


Hình 3. 3 Hình ảnh minh họa cho ảnh từ ứng dụng IP Webcam

Điện thoại được cài đặt chung mạng Wifi với máy tính. Sau đó máy tính sẽ lấy hình ảnh từ địa chỉ URL trên Web Server và tiến hành xử lý nhận dạng rồi đưa ra kết quả



### 3.4 Lưu đồ giải thuật cho việc huấn luyện và nhận dạng chữ viết tay.



Hình 3. 4 Lưu đồ giải thuật cho việc huấn luyện và nhận dạng chữ viết tay

Hình ảnh giấy thi sau khi được thu vào bởi Camera sẽ bao gồm cả mặt giấy thi và thông tin nhiễu như mặt bàn, bụi bẩn, nhiễu hạt, nhiễu muối tiêu... Đồng thời ảnh giấy thi bị nghiêng, sẽ dẫn đến việc nhận dạng văn bản gặp khó khăn. Hình ảnh giấy thi sẽ được tiền xử lý với trình tự như sau:

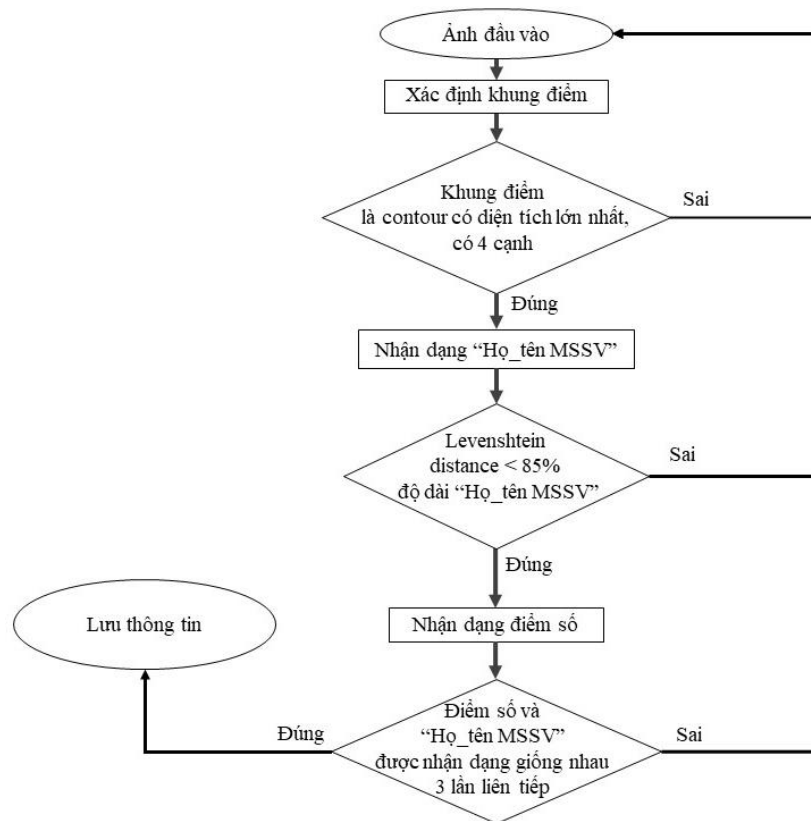
- Hình ảnh được chuyển về ảnh xám vì màu sắc không ảnh hưởng nhiều đến đề tài và đồng thời làm giảm đáng kể chi phí tính toán trong việc xử lý ảnh và nhận dạng.
- Ảnh được phân biệt với mặt bàn dựa vào phương pháp Image Alignment (trích chọn các đặc trưng riêng của giấy thi và so sánh với một tờ giấy thi mẫu). Giấy thi lúc này sẽ được tách ra khỏi nền là mặt bàn và xoay lại cho đúng góc độ.
- Cắt bỏ thủ công nửa dưới của giấy thi nhằm làm giảm chi phí tính toán và tăng tốc độ xử lý. Xác định khung điểm lớn của giấy thi, từ đó giúp xác định vị trí và cắt hình ảnh các thông tin “Họ và tên”, “MSSV”, “Điểm tổng kết” một cách chính xác.

- Tăng độ tương phản với các phép toán xử lý hình thái học là Top Hat và Black Hat nhằm tăng khả năng nhận dạng với những ảnh bị nhòe. Lấy ngưỡng, làm mờ, giảm nhiễu và xóa hàng kẻ ngang (đối với ảnh họ và tên, MSSV) và xóa vòng tròn giới hạn (đối với ảnh điểm số)

Đối với ảnh họ và tên sinh viên, chỉ quá trình tiền xử lý ở trên thôi là chưa đủ, em cần phân tách từ (Word Segmentation) để một lần nữa giảm bớt chi phí tính toán và tăng khả năng nhận dạng họ tên. Có hai cách tiếp cận để xử lý vấn đề. Một là sử dụng thuật toán EAST để tìm các từ viết tay trên toàn ảnh. Sau đó cắt từng từ ra dựa vào vị trí được xác định. Hai là phân tách các từ ngay trên ảnh thu được bằng kỹ thuật Scale Space.

Nhận dạng văn bản: Hình ảnh sẽ được trích chọn đặc trưng để huấn luyện mô hình, mô hình sau khi huấn luyện sẽ được dùng để nhận dạng văn bản trong thực tế. Ở đây em sử dụng hai mô hình riêng biệt để huấn luyện và nhận dạng. Một mô hình nhận dạng họ và tên, một mô hình khác chuyên dùng để nhận dạng MSSV và điểm số. Bộ nhận dạng là mô hình CRNN kết hợp CTC Loss cùng lớp Attention. Trong quá trình làm việc, thông tin đầu ra của lớp Bi – LSTM sẽ được phiên mã Lexicon Based với từ điển là danh sách lớp đầu vào để đưa ra kết quả cuối cùng.

### 3.5 Lưu đồ giải thuật khi nhận dạng trên Video



Hình 3. 5 Lưu đồ giải thuật khi nhận dạng trên Video

Video đầu vào được chia thành các Frame ảnh để nhận dạng, để giảm tải cho quá trình nhận dạng, cứ mỗi 5 Frame sẽ bắt đầu nhận dạng một lần.

Tiếp theo là xác định khung điểm để từ đó trích tách được đúng vị trí những thông tin cần nhận dạng, những tấm hình không xác định được khung điểm, hoặc khung điểm không đáp ứng được các tiêu chí về số cạnh và diện tích sẽ bị hủy và trở về bước đầu tiên

Thông tin “Họ\_và\_tên MSSV” mặc dù có thể được tìm bằng Lexicon Search cho ra độ chính xác khá cao. Tuy nhiên, nếu kết quả sau khi tìm bằng Lexicon Search lại khác biệt quá nhiều so với kết quả Model đưa ra cũng sẽ được cho là không đáng tin cậy. Cụ thể, Levenshtein Distance giữa kết quả gốc của Model và kết quả sau khi tìm bằng Lexicon Search nằm trong khoảng 80% đến 85% là đủ để đảm bảo tính cân bằng giữa tốc độ nhận dạng và độ chính xác

Không giống như thông tin Họ tên và MSSV có sự hỗ trợ lẫn nhau, hay có thể dựa vào danh sách lớp để tăng độ chính xác. Việc nhận dạng điểm số trên từng hình ảnh riêng lẻ lại rất khó khăn vì không có thông tin hỗ trợ để đảm bảo rằng kết quả nhận dạng là đúng hay sai. Nhưng khi nhận dạng với Video thì sẽ có nhiều thông tin hơn ở nhiều Frame ảnh khác nhau. Nếu mô hình nhận dạng được giống nhau cả về thông tin điểm số và “Họ\_tên MSSV” liên tiếp trên 3 lần, khi đó thông tin sẽ được coi là chính xác và không cần cập nhật lại điểm số mỗi khi gặp lại sinh viên đó.

Sau khi đã nhận dạng được văn bản chứa các thông tin của học sinh. Các thông tin ấy sẽ được lưu vào biến hệ thống chứ không liên tục lưu thông tin vào File Excel. Mỗi khi người dùng bấm nút Save trên giao diện, hệ thống sẽ tìm vị trí để nhập điểm trong danh sách lớp (dựa vào thông tin Họ và tên, MSSV) và nhập điểm số cho tất cả các sinh viên vào danh sách lớp.

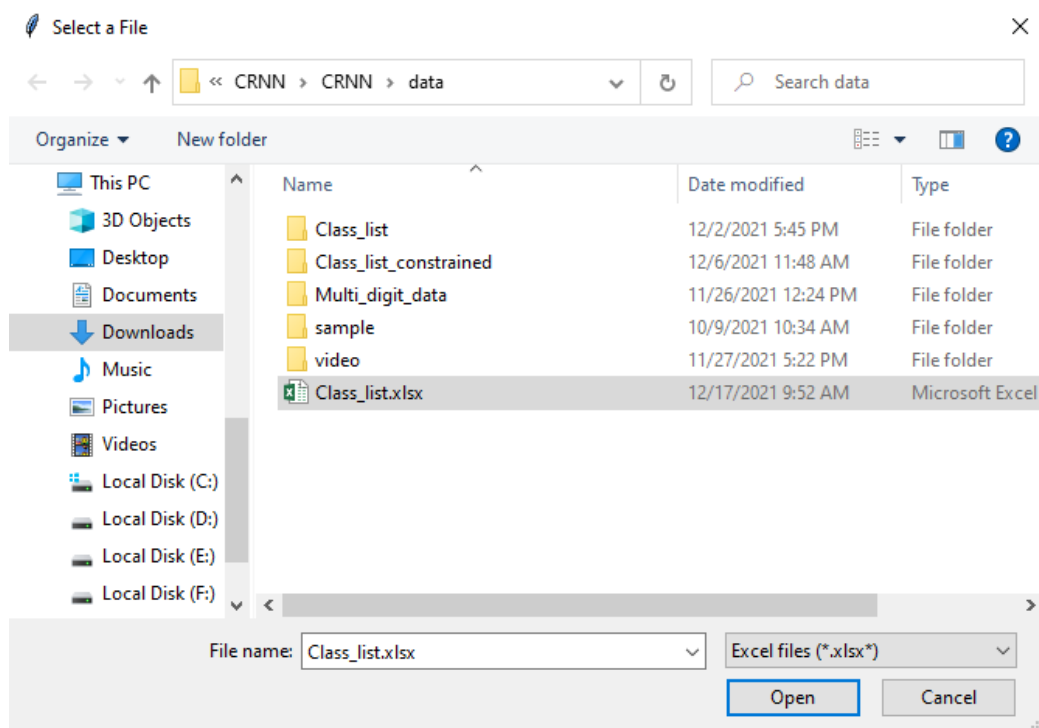
### 3.6 Giao diện người dùng

Hình 3. 6 Giao diện người dùng

Giao diện người dùng trước khi nhận dạng bao gồm những phần sau:

- Text Box để người dùng nhập địa chỉ URL của Camera trên Web Server
- Nút nhấn Browser Files để lấy đường dẫn đến File Excel danh sách lớp có trong máy tính.
- Nút nhấn Start để bắt đầu quá trình nhận diện
- Nút nhấn Save để ghi lại thông tin vào danh sách lớp
- Nút nhấn Quit để thoát khỏi chương trình

Sau khi nhấn nút Browser Files sẽ hiện ra hộp thoại để người dùng có thể chọn File danh sách lớp. Người dùng có thể nhấp chuột vào Combo Box ở góc dưới bên phải của hộp thoại để chọn hiển thị tất cả các File (All Files) hoặc chỉ hiển thị các File Excel. Sau đó người dùng nhấn đúp chuột vào File Excel danh sách lớp cần làm việc.



**Hình 3. 7 Hộp thoại hiển thị sau khi bấm nút Browser Files**

Sau khi chọn danh sách lớp, đường dẫn sẽ hiện ra bên cạnh nút Browser Files để người dùng tiện theo dõi, địa chỉ URL của Camera khi nhập vào cũng có dạng như sau:

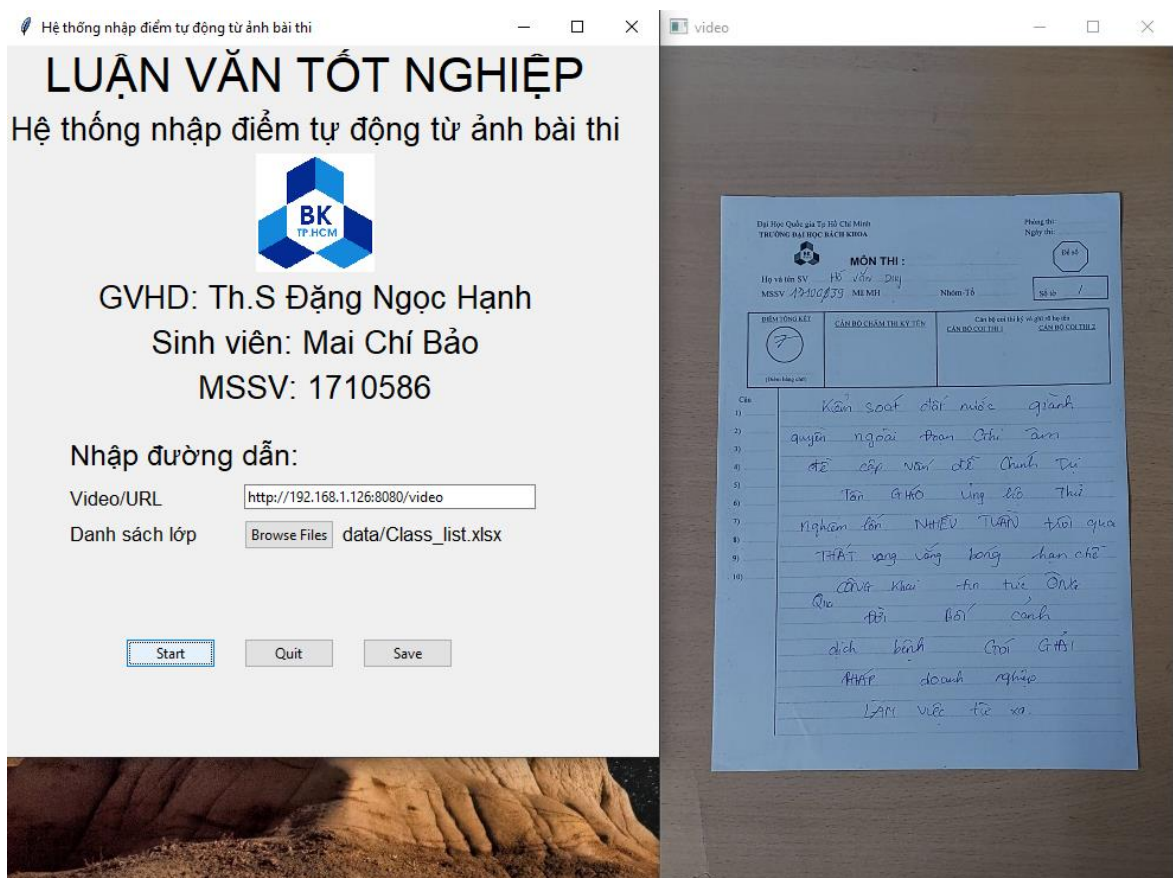
**Nhập đường dẫn:**

Video/URL

Danh sách lớp

Hình 3. 8 Giao diện sau khi nhập đường dẫn

Tiếp theo người dùng nhấn nút Start để bắt đầu làm việc. Hình ảnh giấy thi sẽ liên tục được hiển thị trên màn hình bên cạnh giao diện chính để theo dõi. Như hình dưới đang biểu diễn quá trình nhận dạng giấy thi “Hò Văn Duy 1710039”.



Hình 3. 9 Giao diện hiển thị trong quá trình làm việc

Trên Terminal của hệ thống và hình ảnh Video cũng sẽ hiện thông tin kết quả nhận dạng Họ tên, MSSV và điểm số (không kể đúng sai). Sau khi kết quả nhận dạng được giống nhau 3 lần liên tục thì Terminal và Video sẽ hiển thị dòng trạng thái “Điểm số đã được cập nhật cho Họ\_ và \_tên MSSV là: Điểm\_số”

```
Tên_MSSV: Hồ Văn Duy 1710039
Điểm số: 7.0
Điểm số đã được cập nhật cho Hồ Văn Duy 1710039: 7.0
Điểm số đã được cập nhật cho Hồ Văn Duy 1710039: 7.0
```

Hình 3. 10 Giao diện hiển thị trên Terminal

Hình 3. 11 Hình ảnh hiển thị kết quả nhận dạng gồm điểm số và MSSV

## CHƯƠNG 4. KẾT QUẢ VÀ ĐÁNH GIÁ

### 4.1 Cơ sở dữ liệu

#### 4.1.1 Bộ dữ liệu HANDS-VNOnDB2018

HANDS – VNOnDB2018 (ICFHR2018 Competition on Vietnamese Online Hand written Text Recognition Database) là bộ dữ liệu cung cấp 1.146 đoạn văn bản viết tay bằng tiếng Việt gồm 7.296 dòng, hơn 480.000 nét và hơn 380.000 kí tự được viết bởi 200 người Việt Nam.

Vì bộ dữ liệu này là các từ ở dạng File .inkml nên sẽ được xử lý để trở thành ảnh .PNG kèm nhãn (Ground Truth) phục vụ cho việc nhận dạng.

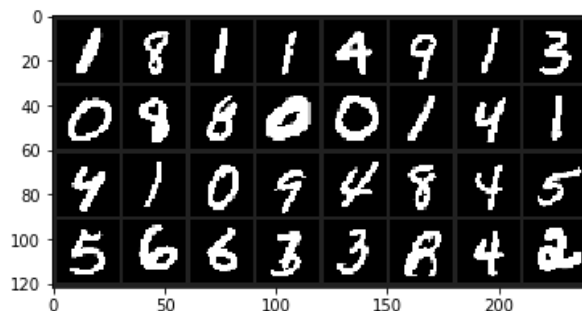
Đây là bộ dữ liệu em dùng để huấn luyện mô hình nhận dạng Họ và tên.



Hình 4. 1 Ảnh chữ viết tay trong bộ dữ liệu HANDS – VNOnDB2018

#### 4.1.2 Bộ dữ liệu MNIST

Trong bộ dữ liệu này, mỗi hình là một ảnh đen trắng chứa một số được viết tay từ 0 đến 9 có kích thước là 28x28. Bộ dữ liệu vô cùng đồ sộ với khoảng 60.000 ảnh dữ liệu huấn luyện, 10.000 ảnh dữ liệu kiểm thử và được sử dụng phổ biến trong các thuật toán nhận dạng ảnh. Đây là bộ dữ liệu em dùng để tự sinh chuỗi số phục vụ cho việc huấn luyện và nhận dạng MSSV và điểm số.



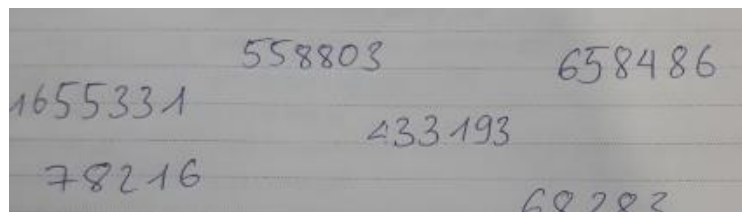
Hình 4. 2 Ảnh mẫu MNIST



### 4.1.3 Bộ dữ liệu thực (tự thu gom)

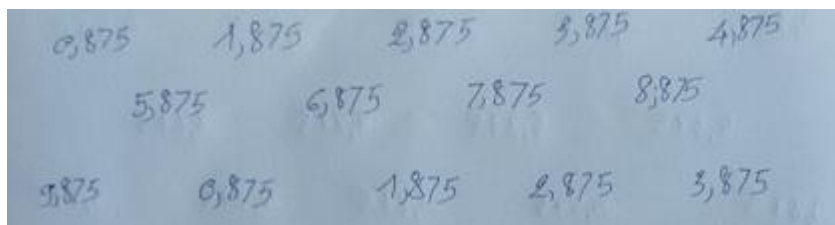
#### 4.1.3.1 Bộ dữ liệu huấn luyện

Gồm 799 hình ảnh số thực được viết tay bởi 6 người. Trong đó gồm 224 ảnh chuỗi số có độ dài từ 3 đến 9 chữ số nhằm giúp tăng cường việc nhận dạng MSSV. Các số được viết trên giấy có hàng kẻ ngang. Sau đó cũng áp dụng các phương pháp tiền xử lý với MSSV để trích tách các hình ảnh



Hình 4. 3 Dữ liệu huấn luyện MSSV

Có 575 ảnh điểm số từ 0 đến 10 với độ chia nhỏ nhất là 0.125 nhằm tăng cường khả năng nhận dạng điểm số. Các hình ảnh cũng được viết trên giấy A4, sau đó sử dụng các phương pháp tiền xử lý cho điểm số để trích tách hình ảnh huấn luyện



Hình 4. 4 Dữ liệu huấn luyện điểm số

#### 4.1.3.2 Bộ dữ liệu kiểm tra

Bao gồm 204 giấy thi được viết tay bởi 10 người. Được chia thành 3 tập con khác nhau nhằm đánh giá các khả năng khác nhau.

##### a) Tập ảnh 122

Là 122 hình ảnh giấy thi mang thông tin “Mai Chí Bảo 1710586” được chụp ở các vị trí, góc máy, ánh sáng khác nhau. Những hình ảnh này được tạo ra chỉ từ một giấy thi nhằm đánh giá khả năng nhận dạng của mô hình với từng sửa đổi trong quá trình huấn luyện và tiền xử lý.

b) Tập 103 ảnh hạn chế

Là 103 ảnh giấy thi khác nhau của 103 sinh viên có trong danh sách lớp đã cho. Kiểu viết và cách viết bị hạn chế (chỉ có thể viết in hoa chữ cái đầu, điểm số không được vượt ra khỏi khung tròn, các số phải có kiểu viết nhất định, rõ nét, đảm bảo khoảng cách giữa các chữ...). Đồng thời phong nền, góc máy, ánh sáng phải đảm bảo các điều kiện nhận dạng, nhằm đánh giá khả năng tiền xử lý và nhận dạng, độ chính xác khi sử dụng.

Hình 4. 5 Ảnh kiểm tra bị hạn chế

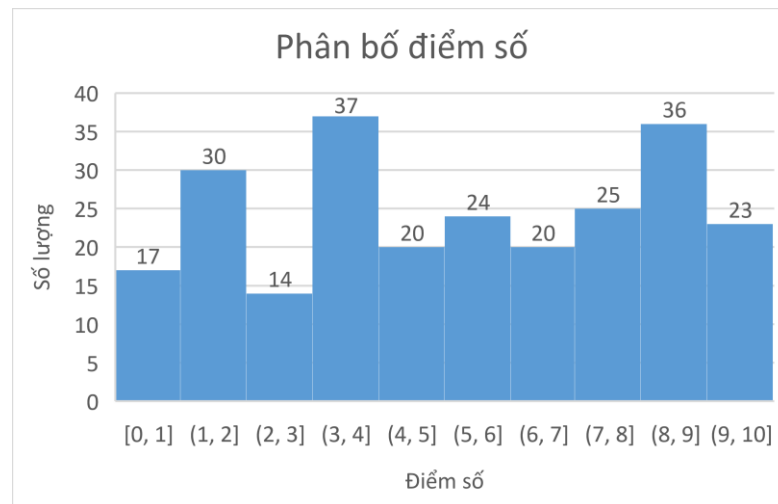
c) Tập 100 ảnh không hạn chế

Là 100 ảnh giấy thi khác nhau của 100 sinh viên có trong danh sách lớp đã cho. Kiểu viết và cách viết không bị hạn chế (có thể viết chữ hoa, in hoa toàn bộ chữ cái, điểm số có thể vượt ra khỏi khung tròn...). Đồng thời phong nền, góc máy, ánh sáng có thể khác nhau, nhằm đánh giá khả năng tiền xử lý và nhận dạng, độ chính xác khi gửi trả về kết quả cuối cùng với những điều kiện bất lợi.

Hình 4. 6 Ảnh kiểm tra không bị hạn chế

d) Danh sách lớp

Danh sách lớp tự tạo có Họ và tên, MSSV, điểm số của 245 sinh viên. Điểm số được tạo bằng hàm Random của Excel, phục vụ cho việc đo đạc kết quả. Điểm số nằm trong khoảng từ 0 đến 10 với độ chia nhỏ nhất là 0.125



Biểu đồ 4. 1 Phân bố điểm số trong danh sách lớp 245 sinh viên

Class_list.xlsx - Excel (Product Activation Fa								
File Home Insert Page Layout Formulas Data Review View Tell me what you want to do...								
I6								
	A	B	C	D	E	F	G	H
1	MSSV	Họ	Tên	TN	diem	diem_test		
2	1710550	Phan Thanh	Ân	59	7.375			
3	1510120	Lê Hoàng	Ân	51	6.375			
4	1610137	Nguyễn Hoàng	Ân	48	6			
5	1710010	Ta Đức	Anh	35	4.375			

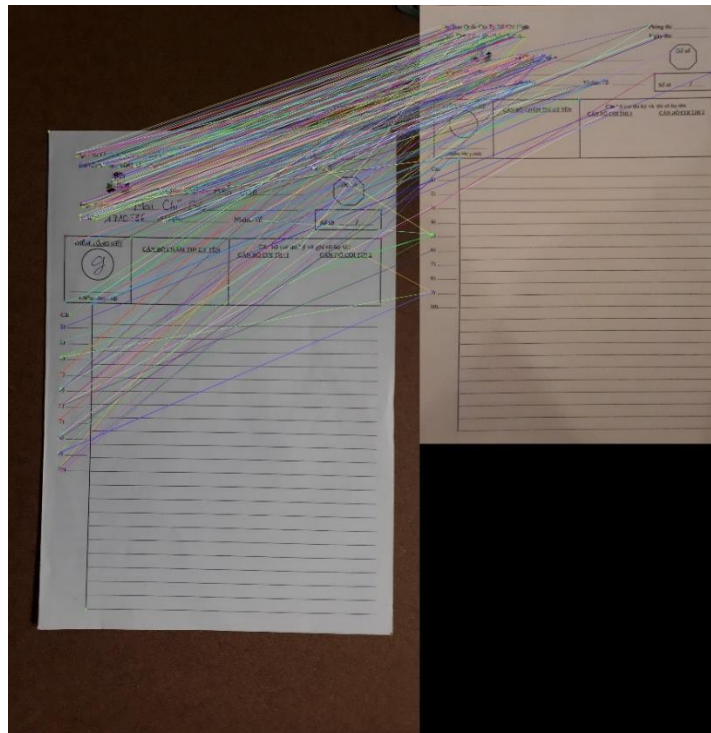
Hình 4. 7 Danh sách lớp File Excel

## 4.2 Tiền xử lý

Ảnh đầu vào sẽ có cả mặt giấy thi và thông tin nhiễu (như mặt bàn, bụi bẩn, nhiễu hạt, nhiễu muối tiêu...). Đồng thời ảnh bị nghiêng, sẽ dẫn đến việc nhận dạng văn bản bị sai và gặp một số bất lợi khác như việc thuật toán EAST chỉ tạo được hộp thoại nằm ngang với Python.



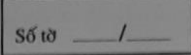

Em đã sử dụng OpenCV để thực hiện các thao tác loại bỏ các thông tin thừa, nhiễu, chuyển về ảnh xám. Sau đó dùng kỹ thuật Image Alignment nhờ vào phương pháp

Feature – Based là kỹ thuật tìm các đặc trưng của ảnh cần hiệu chỉnh rồi sau đó sắp xếp lại ảnh theo một trật tự cho trước.



Hình 4. 8 Ảnh sau khi dùng kỹ thuật Image Alignment

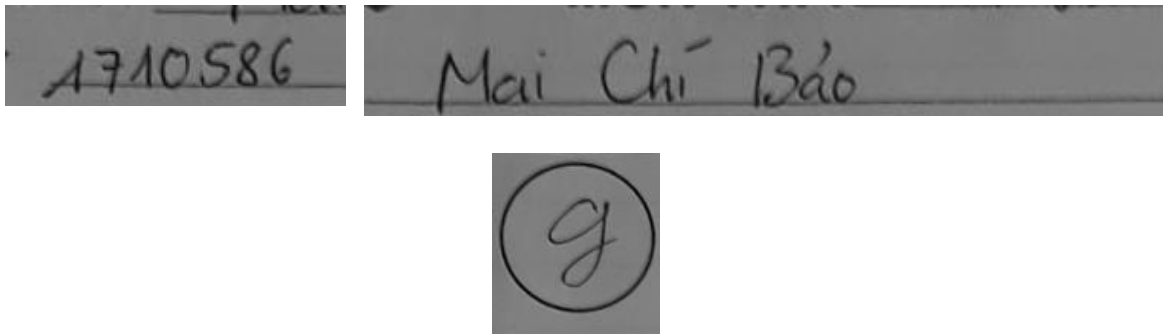
Vì chỉ sử dụng thông tin ở nửa trên của mặt giấy, nên em sẽ cắt bỏ thủ công nửa dưới giấy thi, giữ lại nửa trên với kích thước cố định là 1213x752 Pixels để phục vụ cho các bước xử lý tiếp theo.

Đại Học Quốc Gia Tp.Hồ Chí Minh TRƯỜNG ĐẠI HỌC BÁCH KHOA		Phòng thi: _____ Ngày thi: _____
 <b>MÔN THI :</b> <u>Ảnh văn</u>		
Họ và tên SV <u>Mai Chí Bảo</u> MSSV <u>1710.586</u> Mã MH _____ Nhóm-Tổ _____		
<p>X,Y</p> <p><b>ĐIỂM TỔNG KẾT</b></p> <p style="text-align: center;"></p> <p>(Điểm bằng chữ)</p>	<p><b>CÁN BỘ CHẤM THI KÝ TÊN</b></p> <p>Cán bộ coi thi ký và ghi rõ họ tên</p> <p><b>CÁN BỘ COI THI 1</b>      <b>CÁN BỘ COI THI 2</b></p>	

Hình 4. 9 Ảnh sau khi xác định khung điểm

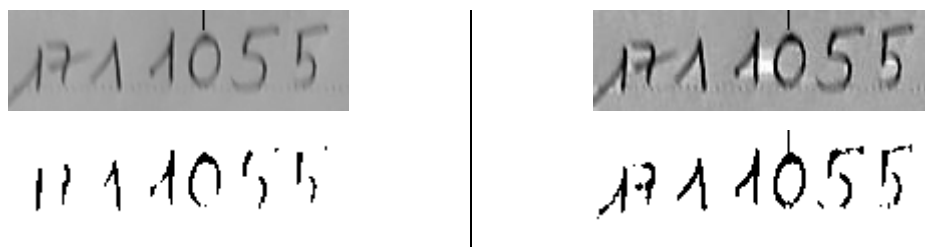
Sau đó em thực hiện xác định khung điểm tổng kết nhờ vào thuật toán tìm Contour của OpenCV. Từ vị trí khung điểm đã có tọa độ (x,y) được đánh dấu đỏ như hình trên. Em tiến hành cắt các khung thông tin với số Pixels cố định từ tọa độ (x,y).

- Ảnh MSSV với kích thước 170x50 Pixels
- Ảnh họ và tên với kích thước 550x55 Pixels
- Ảnh điểm số với kích thước 110x120 Pixels



Hình 4. 10 Ảnh các thông tin sau khi được cắt

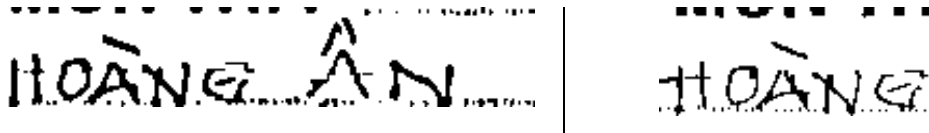
Những ảnh này sau đó được tăng độ tương phản, em sử dụng hai phép Top Hat và Black Hat. Ý tưởng chung là ảnh đầu ra sẽ là ảnh gốc cộng thêm ảnh qua phép Top Hat và trừ đi ảnh qua phép Black Hat. Những chi tiết đã sáng sẽ sáng hơn nhờ phép Top Hat và những chi tiết tối lại càng tối hơn nhờ phép Black Hat. Từ đó sẽ làm tăng độ tương phản cho hình ảnh. Bước này giúp giữ lại nhiều đường nét của ảnh gốc bị nhòe sau khi lấy ngưỡng, nâng cao khả năng nhận dạng.



Hình 4. 11 Trước và sau khi tăng độ tương phản

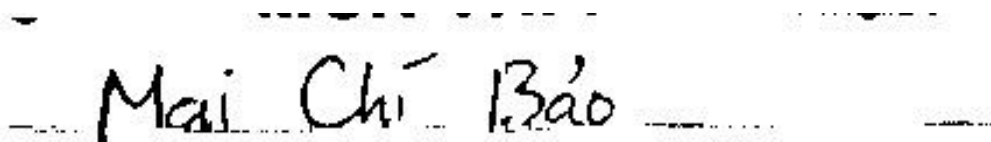
Tiếp theo em lấy ngưỡng nhị phân OTSU (OTSU Threshold), tuy không thể chống chọi tốt với điều kiện ánh sáng bên ngoài bằng ngưỡng nhị phân động (Adaptive Threshold), nhưng đối với ảnh cắt rất nhỏ của thông tin so với toàn giấy thi thì sự ảnh

hưởng của ánh sáng là không đáng kể. Chưa kể việc lấy ngưỡng động dù cho ra ảnh có nét chữ đậm hơn nhưng kèm theo đó là nhiều nhiễu được giữ lại dù đã qua bước làm mờ, giảm nhiễu (Blur). Đặc biệt là dòng chấm ngang dưới làm cho bước phân tách từ (Word Segmentation) không thực sự hiệu quả. Ở đây không có phương pháp nào thực sự tốt hơn, chỉ có sự đánh đổi, điều chỉnh lại thông số các bước cũng như việc chuẩn bị dữ liệu huấn luyện sẽ khác nhau ở cả hai phương pháp.

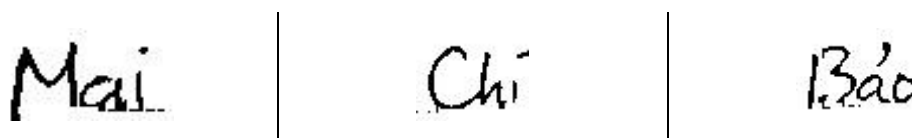


Hình 4. 12 Adaptive Threshold (trái) và OTSU Threshold (phải)

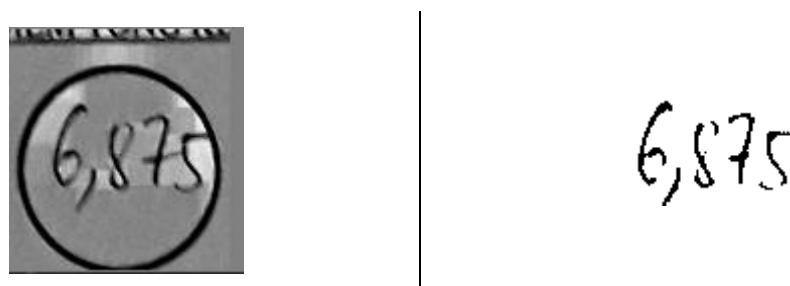
Cuối cùng là xóa hàng kẻ ngang, đối với ảnh điểm số thì em xóa vòng tròn bao quanh, phân tách từ, làm mờ đi những đốm nhiễu và điều chỉnh lại kích thước ảnh để làm đầu vào cho mô hình nhận dạng.



Hình 4. 13 Ảnh họ tên đã được lấy ngưỡng, xóa hàng kẻ và giảm nhiễu



Hình 4. 14 Ảnh Họ và tên sau khi được phân tách từ

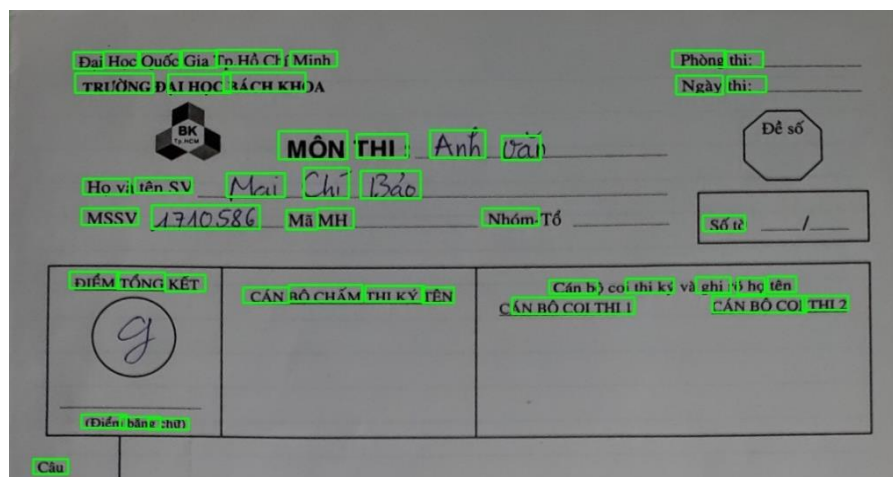


Hình 4. 15 Ảnh trước và sau khi xóa đường tròn bao quanh

### 4.3 Thuật toán EAST và kỹ thuật Scale Space

Thuật toán EAST chỉ làm việc tốt nhất với ảnh có kích thước là bội số của 32. Nên em cần điều chỉnh lại kích thước của ảnh trước khi làm việc. Đồng thời, thực nghiệm cho thấy kích thước ảnh đầu vào có ảnh hưởng lớn đến độ chính xác. Ở đây em thử điều chỉnh ảnh từ kích thước gốc: 1213x752 Pixels lại thành ảnh 640x640 Pixels và ảnh 1024x640 Pixels

EAST dò văn bản khá tốt, cả chữ in và chữ viết tay và có khả năng tách từ trong đoạn văn. Tuy nhiên vẫn có những ô Bounding Box chứa nhiều hơn một từ. Hoặc vị trí đã xác định đúng những từ bị thiếu kí tự hoặc dư từ (nhiều từ cùng nằm trong một Bounding Box).

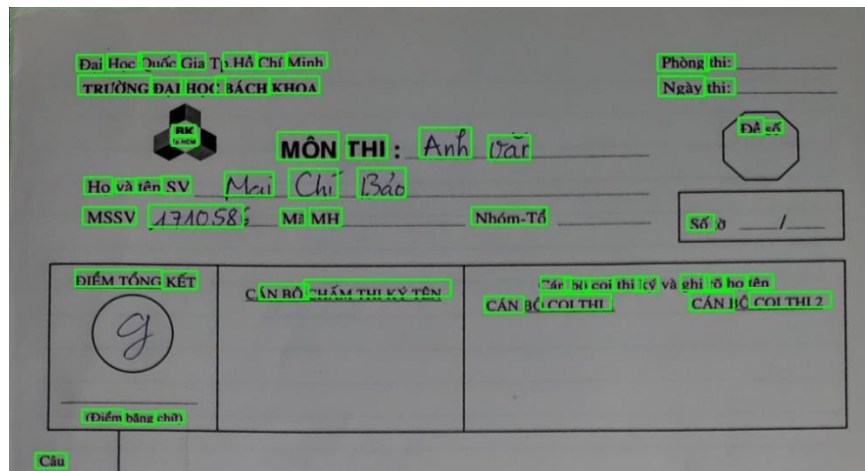


Hình 4. 16 Đầu ra EAST với ảnh đầu vào có kích thước 640x640 Pixels

Khi chuyển về ảnh kích thước 640x640 Pixels để làm việc, thời gian EAST dò văn bản nhanh hơn với 2.1 giây. Dò ra được ít vị trí văn bản hơn với 47 Box và việc tách từ chưa thực sự tốt. Tuy nhiên các thông tin quan trọng ảnh hưởng đến đề tài này là “Họ và tên SV”, “MSSV”, “Điểm tổng kết” được dò ra khá chính xác.

```
origH 752    origW 1213
newH 640     newW 640
[INFO] text detection took 2.104524 seconds
so luong boxes 47
```

Hình 4. 17 Kết quả EAST 1



Hình 4. 18 Đầu ra EAST với ảnh đầu vào có kích thước 1024x640 Pixels

Ở ảnh đầu vào với kích thước 1024x640 Pixels thì tìm được nhiều vị trí văn bản hơn. Các từ có vẻ như được tách chính xác hơn. Tuy nhiên ở vị trí “MSSV” đã lấy thiếu mất số 6. Thời gian thực hiện dò văn bản cũng chậm hơn nhiều là 3.85 giây.

```
origH 752    origW 1213
newH 640    newW 1024
[INFO] text detection took 3.854653 seconds
so luong boxes 58
```

Hình 4. 19 Kết quả EAST 2

Tiếp theo em thử nghiệm phân tách các từ trong ảnh họ và tên bằng cả kỹ thuật Scale Space và EAST.

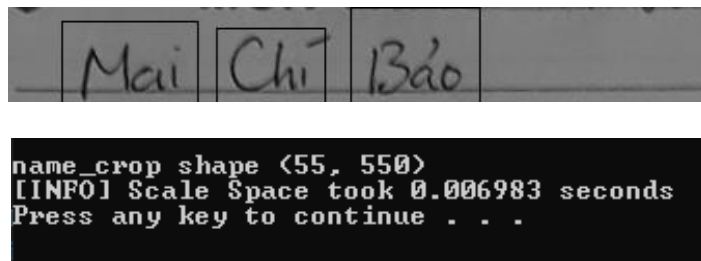
```
name_crop shape (55, 550, 3)
origH 55    origW 550
newH 64    newW 576
[INFO] text detection took 0.315124 seconds
so luong boxes 2
[INFO] EAST took 2.160688 seconds
```

EAST

Hình 4. 20 Kích thước ảnh, thời gian và kết quả thực hiện bằng EAST

Thuật toán EAST cho kết quả đầu ra không chính xác (mất chữ “Chí”) và thời gian thực hiện rất chậm so với kỹ thuật Scale Space (0.007 giây so với 2.16 giây). EAST có thể phát hiện văn bản phi cấu trúc với độ chính xác và tốc độ cao, tuy nhiên với đề tài này thì kỹ thuật Scale Space cho kết quả khả quan hơn.





Hình 4. 21 Kết quả khi thực hiện phân tách từ bằng kỹ thuật Scale Space

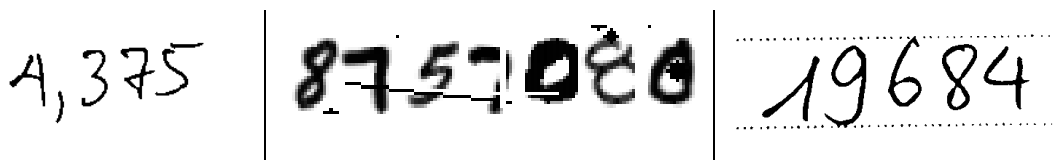
#### 4.4 Nhận dạng với mô hình CRNN kết hợp CTC Loss

##### 4.4.1 Thông số cài đặt

Các ảnh đầu vào này được Resize và Padding nếu cần để có cùng kích thước là 32x128 Pixels, sau đó được Transpose và chuẩn hóa giá trị để đưa vào Model.

Model được sử dụng có lớp CNN từ VGG16 gồm 7 lớp Convolution và 5 lớp Pooling xen kẽ, kết hợp 2 lớp Batch Normalization nhằm chuẩn hóa dữ liệu với những Model phức tạp, tránh Bias. Sử dụng 2 lớp Bi – LSTM 256 Unit ẩn kèm theo Drop Out 25% để tránh Overfitting. Ngoài ra còn kết hợp các phương pháp khác:

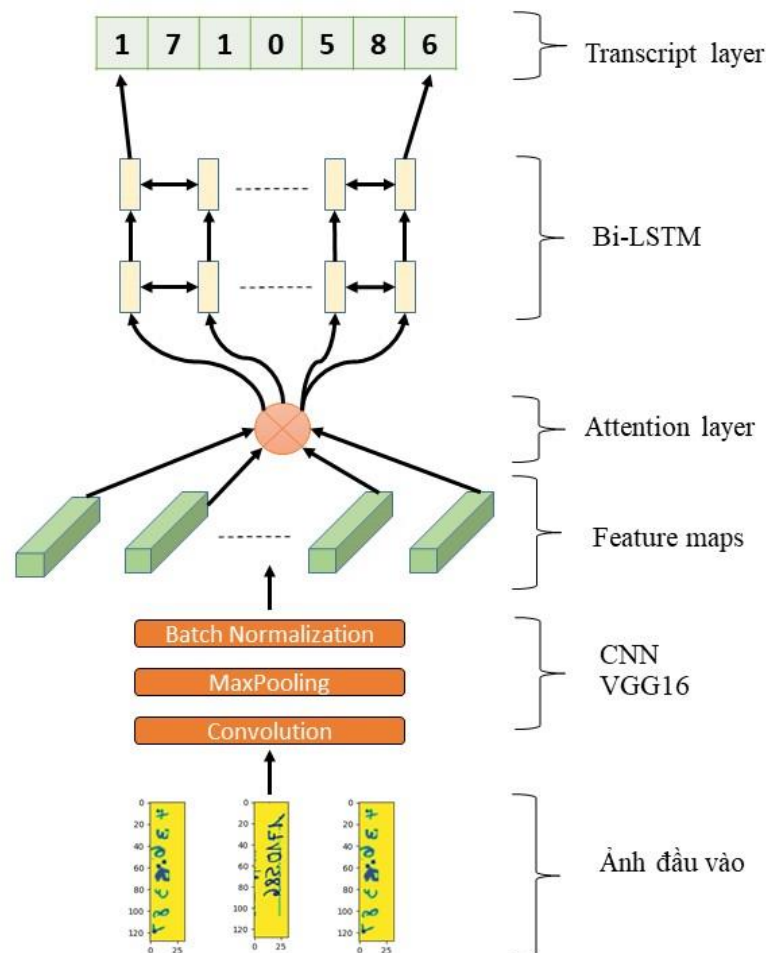
- Lớp Attention được chứng minh khi kết hợp với CTC Loss có thể vừa làm tăng khả năng nhận dạng, vừa giảm thời gian huấn luyện.
- Data Augmentation: thêm Noise, giãn nở, co chữ kết hợp với các phương pháp khác như Elastics Transformation, thay đổi tỉ lệ (Scale), xoay ảnh (Rotate), làm giảm khoảng cách giữa các số, Random Cutout dọc và ngang, thêm đường kẻ...



Hình 4. 22 Ảnh chuỗi số huấn luyện

- Thay đổi Learning Rate 2 lần từ 0.001 xuống 0.00001 theo số Epoch sẽ dễ hội tụ tại điểm cực trị hơn. Tuy nhiên, việc thay đổi ở Model họ tên chỉ có tác dụng ở lần chuyển đầu tiên, càng chuyển nhiều lần thì tốc độ thay đổi càng chậm và Model cho thấy dấu hiệu bị Overfitting. Ở pha huấn luyện cho Model số thì việc thay đổi Learning Rate không có nhiều tác dụng

- Early Stopping nhằm tránh Overfitting, thường được đặt ở 10 – 15 Epochs



Hình 4. 23 Model nhận dạng được sử dụng trong đề tài

Thông số cài đặt chính cho cả hai Model như sau:

	Model họ và tên	Model số
Number of Data	110734	102255
Number of Train Data	94123	81804
Numver of Valid Data	16611	20451
Split_Train_valid	0.15	0.2
num_of_characters	148	11
max_str_len	11	10
num_of_timestamps	31	31
batch_size	128	128

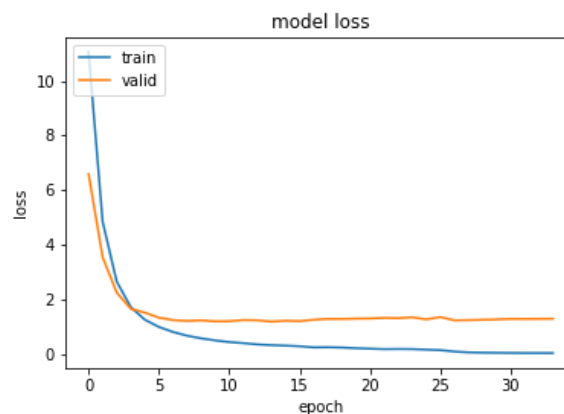
Bảng 4. 1 Thông số cài đặt cho Model họ tên và Model số

#### 4.4.2 Đánh giá trên tập 122 ảnh

Đầu tiên em thử đánh giá với 122 ảnh giấy thi có tên “Mai Chí Bảo” với các điều kiện ánh sáng, góc độ khác nhau để đánh giá độ chính xác trong việc nhận dạng họ tên và MSSV với các thay đổi về dữ liệu và mô hình. Kết quả được đối chiếu với danh sách lớp tự tạo gồm 245 người.

	CRNN + CTC	+ Data Augmentation	+ Learning Rate thay đổi	+ Attention	+ Lexicon search
CER	35.25%	23.40%	16.77%	16.24%	0.45%
WER	74.59%	69.40%	45.63%	47.27%	0.55%

**Bảng 4. 2 Kết quả nhận dạng họ tên trên tập 122 ảnh**

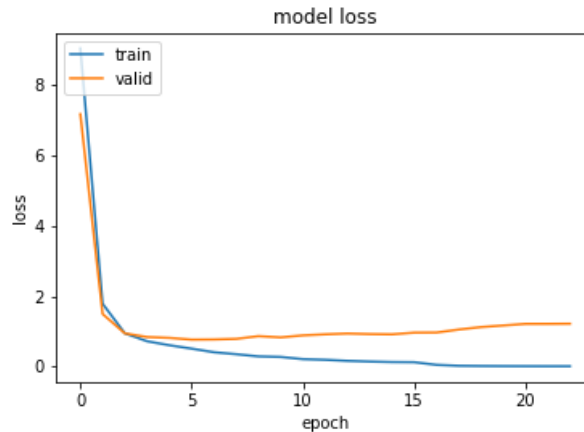


**Hình 4. 24 Quá trình huấn luyện Model họ và tên**

Ở Model nhận dạng số em chỉ thử lại với các tập dữ liệu khác nhau (các phương pháp Data Augmentation khác nhau), trong khi giữ nguyên Model CRNN + CTC, lớp Attention cùng các phương thức khác.

	Data 7 chữ số kèm Blob Noise	+ Data có 1,3,4,5,7 chữ số	+ Rotate, scale từng chữ số, Random Cutout, Thêm Line Noise	+ Thay đổi khoảng cách giữa các số, Scale, Rotate toàn ảnh	+ Thêm Data thật (799 ảnh)	+ Lexicon Search
CER	63.82%	48.24%	45.55%	13.58%	3.63%	2.58%
WER	100.00%	100.00%	100.00%	63.11%	22.95%	13.11%

**Bảng 4. 3 Kết quả nhận dạng MSSV và điểm số trên tập 122 ảnh**



Hình 4. 25 Quá trình huấn luyện Model MSSV

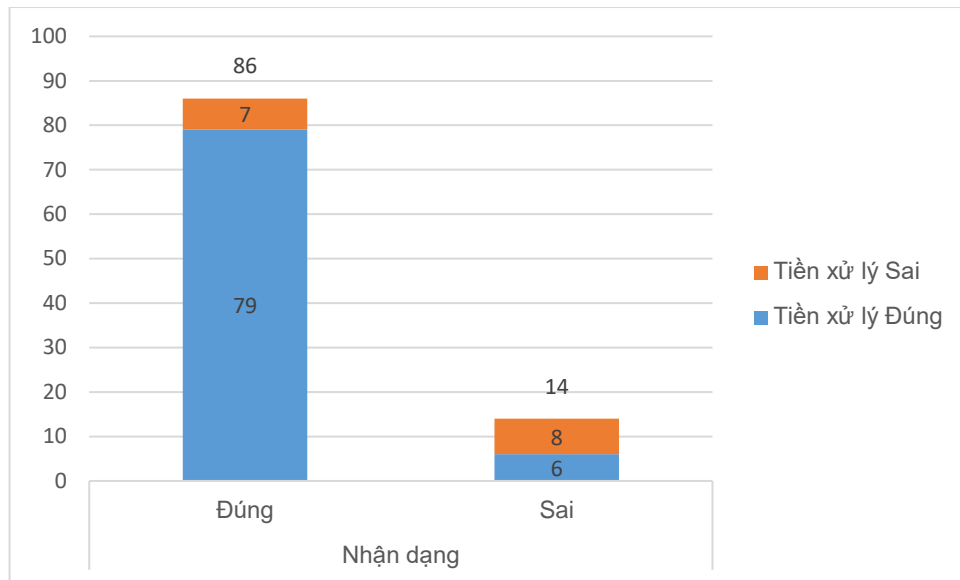
Thời gian nhận dạng nằm trong khoảng từ 1.6 đến 2 giây cho một ảnh. Sau khi thực hiện Lexicon Search theo kiểu kết hợp để tìm số thứ tự sinh viên có dạng “Họ\_và\_tên MSSV” (VD: “Mai Chí Bảo 1710586”) thì CER và WER đều đạt 0%. Đồng nghĩa với độ chính xác khi tìm số thứ tự sinh viên đạt 100%

#### 4.4.3 Đánh giá trên tập 100 ảnh không hạn chế

Em tự tạo 100 ảnh giấy thi không bị hạn chế về kiểu chữ, cách viết, phông nền...với họ tên, MSSV và điểm số khác nhau của các sinh viên khác nhau trong tập danh sách lớp gồm 245 sinh viên. Sau đó đánh giá khả năng tìm số thứ tự của sinh viên “Họ\_và\_tên MSSV” và khả năng nhận dạng đúng điểm số

##### 4.4.3.1 Nhận dạng họ tên và MSSV

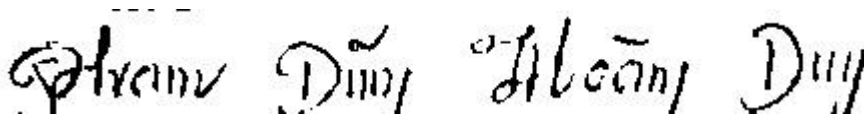
Hình ảnh tiền xử lý đúng là hình ảnh được cắt đúng vị trí thông tin, phân tách từ đúng, không có quá nhiều nhiễu, còn chứa lại đủ thông tin để là con người cũng có thể nhận dạng.



**Biểu đồ 4. 2 Kết quả nhận dạng số thứ tự sinh viên trên tập 100 ảnh không hạn chế**

Như vậy, việc trích tách, tiền xử lý thông tin họ tên và MSSV cho độ chính xác lên đến 85%. Tuy nhiên không có ảnh nào trong 100 ảnh bị cắt sai vị trí của Họ tên và MSSV. Lỗi đa phần ở bước phân tách từ sai. Khả năng trả về đúng số thứ tự của sinh viên lên đến 86%, nhưng khả năng nhận dạng đúng trên tập ảnh được tiền xử lý đúng đạt 92.94%

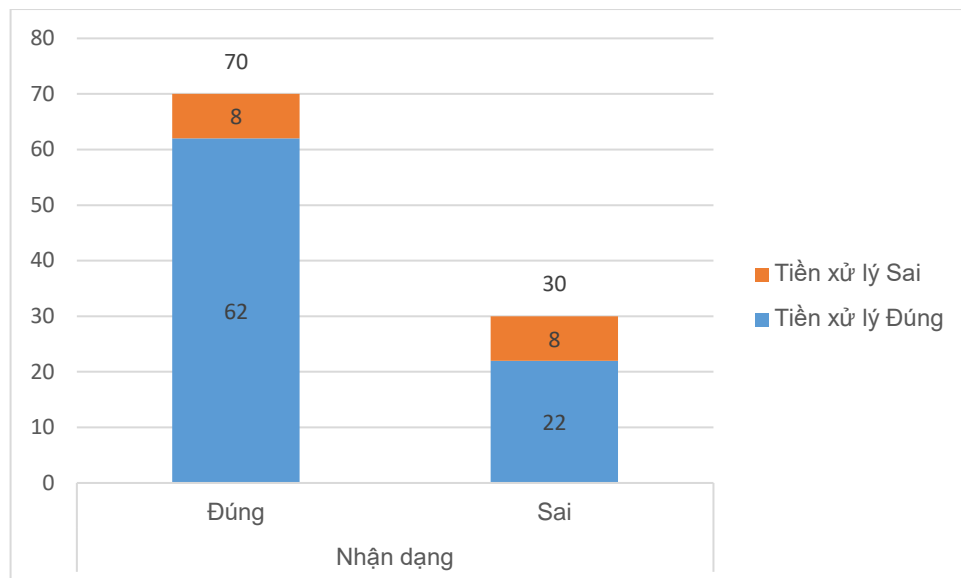
Một số lỗi nhận dạng ở đây không đáng kể như kiểu chữ không xuất hiện trong Data Train (kiểu chữ viết hoa), hay Model bị lẫn lộn giữa các thông tin sinh viên gần như là giống nhau “Nguyễn Đức Hoàng 1711393” và “Nguyễn Đắc Hoàng 1711392”. Cho nên độ chính xác này là chấp nhận được và có khả năng cải thiện thêm nếu như áp dụng thêm một số hạn chế cho người sử dụng (chỉ viết chữ in hoa, không tẩy xóa, chữ đẹp, rõ nét...)



**Hình 4. 26 Ảnh nhận dạng lỗi vì không có dữ liệu trong Data Train**

#### 4.4.3.2 Nhận dạng điểm số

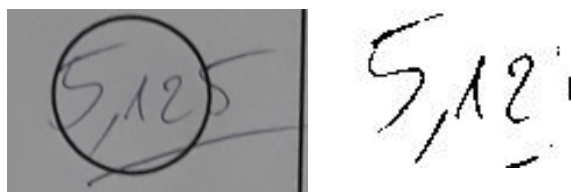
Ở đây ảnh nhận dạng sai là ảnh cắt sai vị trí, ảnh nhiễu nhiều, không còn chứa đủ thông tin để có thể nhận dạng.



**Biểu đồ 4. 3 Kết quả nhận dạng điểm số của sinh viên trên tập 100 ảnh không hạn chế**

Như vậy khả năng tiền xử lý đúng cho ô điểm số là 84%. Khả năng trả về điểm số đúng là 70%. Trong đó nhận dạng đúng điểm số trên những hình đã tiền xử lý đúng là 73.81%

Lý do chính cho lỗi tiền xử lý là chữ số được giáo viên viết quá to, vượt ra khỏi đường tròn giới hạn dẫn đến lúc cắt bị mất thông tin



**Hình 4. 27 Lỗi điểm số vượt ra khỏi vòng tròn giới hạn**

Đồng thời có thể vì quá trình cắt thông tin bị sai lệch (cụ thể là sai từ bước Image Alignment), làm cho vòng tròn bị hở, từ đó khó xác định Contour để xóa vòng tròn hoặc hệ thống nhầm lẫn và xóa hết thông tin điểm số.



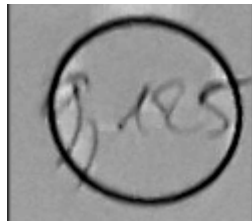
**Hình 4. 28 Lỗi cắt sai vị trí ảnh. (Contour được xác định có màu đen đậm)**



,75

**Hình 4. 29 Lỗi vẽ contour (Contour được xác định có màu đen đậm)**

Hoặc ảnh đầu vào bị nhòe thì sau quá trình lấy ngưỡng cũng có thể làm mất hết thông tin ảnh



**Hình 4. 30 Lỗi ảnh nhòe gây mất thông tin**

Một số lỗi nhận dạng phổ biến là bị nhầm lẫn giữa số 9 và 2, số 9 và 0, số 7 và 1, số 7 và 2, số 3 và 5 nhưng chủ yếu là do chữ viết rất xấu, hoặc kiểu chữ ít khi xuất hiện trong tập Train. Đồng thời dấu “,” hay dấu gạch dưới cũng làm Model nhận dạng sai.



**Hình 4. 31 Dấu “,” làm số 0 nhầm thành 9**

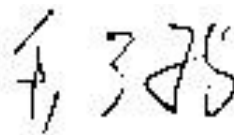


**Hình 4. 32 Số 3 nét ngang không có trong tập Train**



**Hình 4. 33 Chữ số 5 khó nhận dạng, dấu gạch dưới gây nhiễu**

Quá trình tiền xử lý có thể làm mất nét chữ, dẫn đến khó nhận dạng. Tuy nhiên những điều này đã được khắc phục hiệu quả bằng các phương pháp Data Augmentation



**Hình 4. 34 Số 4 bị nhầm thành số 7**

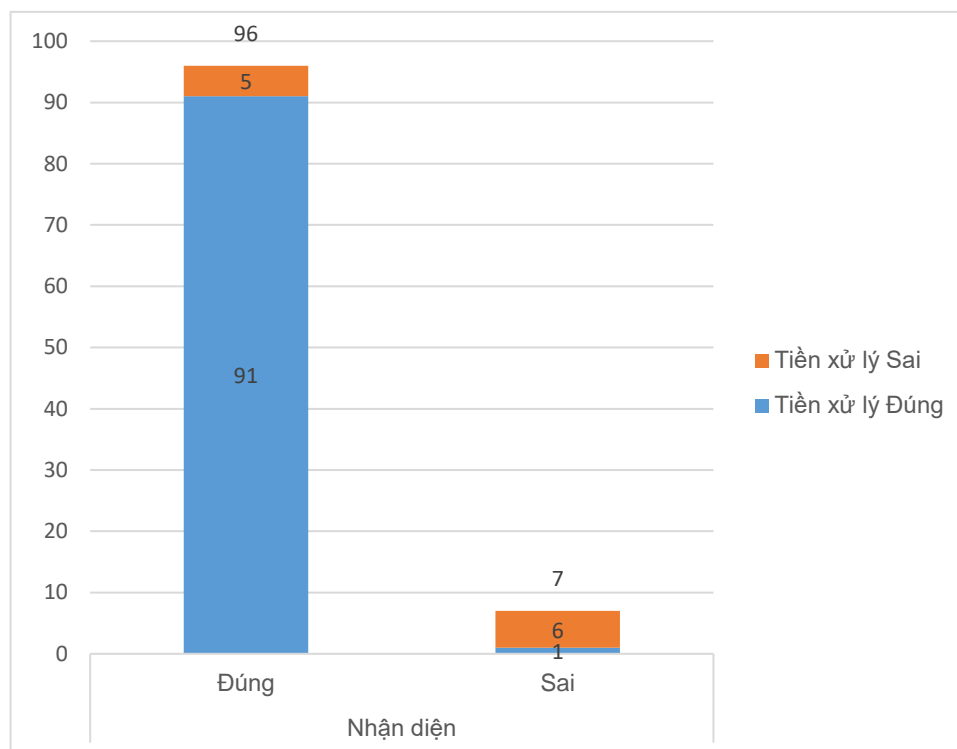
Các lỗi này đều có thể khắc phục được bằng cách thêm dữ liệu các kiểu chữ khác nhau vào quá trình huấn luyện hoặc áp đặt những hạn chế cho người sử dụng. Ví dụ số 7 phải có nét cắt ngang, số 3 tròn vành, số 1 phải có nét chéo lên...

#### 4.4.4 Đánh giá trên tập 103 ảnh hạn chế

Tập 103 ảnh đã bị hạn chế về cách viết, kiểu viết của người sử dụng, đồng thời là cả về vị trí góc chụp của Camera nhằm hạn chế các lỗi phát sinh chủ yếu từ người sử dụng để đánh giá sâu hơn khả năng nhận dạng của hệ thống.

##### 4.4.4.1 Nhận dạng họ tên và MSSV

Ảnh dưới cho thấy việc trích tách tiền xử lý thông tin họ tên và MSSV cho độ chính xác lên đến 89.3%. Khả năng trả về đúng số thứ tự của sinh viên lên đến 93.2%, khả năng nhận dạng đúng trên tập ảnh được tiền xử lý đúng đạt 98.91%

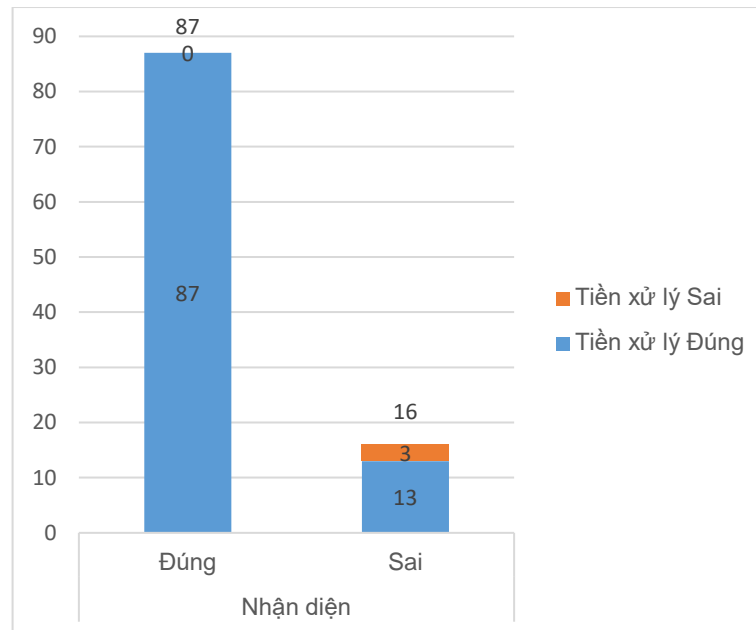


Biểu đồ 4. 4 Kết quả nhận dạng số thứ tự sinh viên trên tập 103 ảnh hạn chế

##### 4.4.4.2 Nhận dạng điểm số

Khả năng tiền xử lý đúng cho ô điểm số là 97.09%. Kết quả trả về điểm số đúng là 84.47%. Trong đó khả năng nhận dạng đúng điểm số trên những hình tiền xử lý đúng là 87%





**Biểu đồ 4. 5 Kết quả nhận dạng điểm số trên tập 103 ảnh hạn chế**

#### 4.4.5 Đánh giá trên Video

Trong thực tế, ảnh được đưa vào xử lý liên tục như Video nên khả năng nhận dạng sẽ khác so với việc nhận dạng riêng lẻ từng ảnh một, đặc biệt là việc nhận dạng điểm số. Vì bây giờ thông tin điểm số sẽ được so sánh liên tục để biết điểm số thu được có đáng tin cậy hay không. Đồng thời các bước tiền xử lý khác, cũng như thông tin nhận dạng “Họ\_và\_tên MSSV” nếu không đảm bảo các tiêu chí nhất định thì hình ảnh đó cũng sẽ bị hủy, góp phần tăng tăng độ tin cậy cho kết quả, cũng như giảm sai số của kết quả cuối cùng.

Sau khi thử nghiệm trên 45 tờ giấy thi bất kì, hệ thống cho ra kết quả cuối cùng bao gồm Họ và tên, MSSV và điểm số với độ chính xác khi nhập điểm là 95.55% (đúng 43/45 giấy thi). Hai giấy thi còn lại không thể đáp ứng được điều kiện cuối cùng (nhận dạng 3 lần liên tiếp giống nhau) để được lưu thông tin vào danh sách lớp.

Hình dưới là ảnh giấy thi của một sinh viên Bách Khoa. Ở phần MSSV có gạch bỏ số 9 nhưng điều này vẫn không làm ảnh hưởng đến kết quả. Hầu hết thông tin “Họ\_và\_tên MSSV” đều được nhận dạng đúng. Về điểm số, tuy có một vài lần nhận dạng sai (từ 7 sang 3). Nhưng cuối cùng kết quả khi nhập điểm vẫn đúng.

Đại Học Quốc gia Tp Hồ Chí Minh  
TRƯỜNG ĐẠI HỌC BÁCH KHOA

Phòng thi: \_\_\_\_\_  
Ngày thi: \_\_\_\_\_

**MÔN THI :** \_\_\_\_\_

Họ và tên SV Hỗ Văn Duy  
MSSV 1710039 Mã MH \_\_\_\_\_ Nhóm-Tổ \_\_\_\_\_

Đề số \_\_\_\_\_  
Số tờ 1

<b>ĐIỂM TỔNG KẾT</b>  (Điểm bằng chữ)	<b>CÁN BỘ CHẤM THI KÝ TÊN</b>  	Cán bộ coi thi ký và ghi rõ họ tên <b>CÁN BỘ COI THI 1</b> <b>CÁN BỘ COI THI 2</b>
---	---------------------------------------	--

Hình 4. 35 Hình ảnh giấy thi mẫu

```

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Điểm số: 7.0

Tên_MSSV: Hỗ Văn Duy 1710039
Điểm số: 3.0

Tên_MSSV: Hỗ Văn Duy 1710039
Điểm số: 7.0

Tên_MSSV: Hỗ Văn Duy 1710039
Điểm số: 7.0

Tên_MSSV: Hỗ Văn Duy 1710039
Điểm số: 7.0
Điểm số đã được cập nhật cho Hỗ Văn Duy 1710039: 7.0
Điểm số đã được cập nhật cho Hỗ Văn Duy 1710039: 7.0

```

Hình 4. 36 Minh họa kết quả nhận dạng trên Terminal

Thời gian nhận dạng trung bình là dưới 1 phút/giấy thi. Độ lệch thời gian từ lúc giáo viên bỏ giấy thi vào đến lúc được hiển thị trên màn hình là từ 10 – 15 giây. Độ trễ thời gian khá lớn. Nhưng ở đề tài này, vì giáo viên chấm bài song song với quá trình nhận dạng của hệ thống và thời gian giữa những lần chấm như vậy khoảng 2 phút. Nên thời gian xử lý như vậy là phù hợp, có thể cân nhắc đánh đổi giữa độ chính xác và tốc độ nhận dạng.

## CHƯƠNG 5. KẾT LUẬN

### 5.1 Tóm tắt và kết luận chung.

#### 5.1.1 Đóng góp của luận văn

Trong quá trình nghiên cứu và xây dựng “Hệ thống nhập điểm tự động từ ảnh bài thi”, luận văn đã đạt được một số nội dung như sau:

- Đề ra được một mô hình tổng quát cho việc nhập điểm số trên giấy thi một cách hoàn toàn tự động từ việc tiền xử lý ảnh, nhận dạng với mô hình Deep Learning cho đến các phương thức làm việc, giao diện người dùng
- Áp dụng mô hình CRNN và CTC Loss để nhận dạng chữ viết tay tiếng Việt
- Tạo ra một nguồn dữ liệu cho việc huấn luyện và kiểm thử mô hình

#### 5.1.2 Hạn chế của luận văn

Ngoài những đóng góp mà luận văn mang lại, vì lý do thời gian cũng như tình hình dịch bệnh Covid – 19 đang diễn ra phức tạp trên khắp cả nước nên luận văn vẫn còn một số hạn chế nhất định.

- Việc thiếu dữ liệu để huấn luyện làm cho mô hình vẫn chưa thực sự đạt kết quả tốt trong việc nhận dạng với từng hình ảnh riêng lẻ
- Mô hình vẫn còn xử lý chậm vì chưa tối ưu hóa được giải thuật và cả những hạn chế về phần cứng của máy tính hay đường truyền Internet
- Khi sử dụng vẫn còn cần giới hạn người dùng trong kiểu viết, cách viết, cách trình bày... để mô hình có thể hoạt động tốt nhất có thể
- Mô hình vẫn chưa thực sự là một công cụ hoàn chỉnh và đầy đủ các chức năng để có thể sử dụng trong thực tiễn. Ví dụ như việc chưa có khả năng kết nối, tải dữ liệu, nhập điểm lên Internet, hệ thống chỉ có khả năng làm việc với các danh sách lớp ở dạng File Excel, giao diện sử dụng còn chưa đẹp mắt.

## 5.2 Hướng phát triển

Từ những hạn chế nêu trên, những hướng mới để phát triển luận văn bao gồm:

- Thay vì tiền xử lý ảnh ở phần đầu của hệ thống, có thể huấn luyện với dữ liệu hình ảnh thực có phong nền. Tuy nhiên điều này cần một bộ dữ liệu cực lớn cũng như thời gian huấn luyện lâu
- Phát triển thêm nhiều tính năng mới. Kết hợp với các bảng điểm Online. Chính sửa lại giao diện sử dụng cho đẹp và phù hợp với mục đích thương mại.
- Tối ưu hóa giải thuật nhằm tăng tốc độ xử lý kể cả trên những Smart Phone hay máy tính có cấu hình phổ thông

## TÀI LIỆU THAM KHẢO

- [1] A. F. M. Agarap, "An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification," *arXiv*, pp. 5–8, 2017.
- [2] S. Ahlawat and A. Choudhary, "Hybrid CNN-SVM Classifier for Handwritten Digit Recognition," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 2554–2560, 2020, doi: 10.1016/j.procs.2020.03.309.
- [3] M. Ahmed and A. Abidi, "Review on Optical Character Recognition," *Irjet*, vol. 06, no. June, pp. 3666–3669, 2019, [Online]. Available: <https://www.irjet.net/archives/V6/i6/IRJET-V6I6736.pdf>.
- [4] J. C. Aradillas Jaramillo, J. J. Murillo-Fuentes, and P. M. Olmos, "Boosting handwriting text recognition in small Databases with transfer learning," *Proc. Int. Conf. Front. Handwrit. Recognition, ICFHR*, vol. 2018-August, pp. 429–434, 2018, doi: 10.1109/ICFHR-2018.2018.00081.
- [5] H. S. M. Beigi, "An Overview of Handwriting Recognition," *Proc. 1<sup>st</sup> Annu. Conf. Technol. Adv. Dev. Ctries.*, no. February 1997, pp. 30–46, 1993.
- [6] A. Choudhary, R. Rishi, and S. Ahlawat, "New character segmentation approach for off-line cursive handwritten words," *Procedia Comput. Sci.*, vol. 17, pp. 88–95, 2013, doi: 10.1016/j.procs.2013.05.013.
- [7] Darmatasia and M. I. Fanany, "Handwriting recognition on form document using convolutional neural network and support vector machines (CNN-SVM)," *2017 5th Int. Conf. Inf. Commun. Technol. ICoICT 2017*, no. April, 2017, doi: 10.1109/ICoICT.2017.8074699.
- [8] M. Elleuch, R. Maalej, and M. Kherallah, "A New design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition," *Procedia Comput. Sci.*, vol. 80, pp. 1712–1723, 2016, doi: 10.1016/j.procs.2016.05.512.
- [9] A. Kaur, S. Baghla, and S. Kumar, "Study of Various Character Segmentation Techniques for Handwritten Off-Line Cursive Words: a Review," *Int. J. Adv. Sci. Eng. Technol.*, no. 3, pp. 2321–9009, 2015.
- [10] H. Khandelwal, S. Gupta, and A. K. Jai, "Review of Offline Handwriting Recognition Techniques in the fields of HCR and OCR," *Int. J. Comput. Trends Technol.*, vol. 47, no. 3, pp. 161–164, 2017, doi: 10.14445/22312803/ijctt-v47p123.
- [11] K. I. Kim, K. Jung, and J. H. Kim, "Texture-Based Approach for Text Detection in Images Using Support Vector Machines and Continuously Adaptive Mean Shift

- Algorithm,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1631–1639, 2003, doi: 10.1109/TPAMI.2003.1251157.
- [12] A. S. Nadarajan and A. Thamizharasi, “A Survey on Text Detection in Natural Images,” vol. 6, no. 1, pp. 60–66, 2018.
- [13] H. T. Nguyen, C. T. Nguyen, P. T. Bao, and M. Nakagawa, “A Database of unconstrained Vietnamese online handwriting and recognition experiments by recurrent neural networks,” *Pattern Recognit.*, vol. 78, pp. 291–306, 2018, doi: 10.1016/j.patcog.2018.01.013.
- [14] A. Nikitha, J. Geetha, and D. S. Jayalakshmi, “Handwritten Text Recognition using Deep Learning,” *Proc. - 5th IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. RTEICT 2020*, pp. 388–392, 2020, doi: 10.1109/RTEICT49044.2020.9315679.
- [15] A. Rehman, “Cursive Overlapped Character Segmentation: An Enhanced Approach,” pp. 1–10, 2019, [Online]. Available: <http://arxiv.org/abs/1904.00792>.
- [16] B. Shi, X. Bai, and C. Yao, “An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, 2017, doi: 10.1109/TPAMI.2016.2646371.
- [17] R. . Thirumalesh. B.S, “International journal of engineering sciences & research technology a comprehensive study on 5,” *Int. J. Eng. Sci.*, vol. 7, no. 7, pp. 247–253, 2018, [Online]. Available: <http://www.ijesrt.com>.
- [18] Y. Wang, W. Xiao, and S. Li, “Offline Handwritten Text Recognition Using Deep Learning: A Review,” *J. Phys. Conf. Ser.*, vol. 1848, no. 1, 2021, doi: 10.1088/1742-6596/1848/1/012015.
- [19] X. Zhou *et al.*, “EAST: An efficient and accurate scene text detector,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 2642–2651, 2017, doi: 10.1109/CVPR.2017.283.