

ĐẠI HỌC QUỐC GIA TP. HCM  
TRƯỜNG ĐẠI HỌC BÁCH KHOA

-----  
**NGUYỄN XUÂN THÀNH**

**PHÁT HIỆN VÀ NHẬN DIỆN BIỂN SỐ XE  
SỬ DỤNG PHƯƠNG PHÁP HỌC SÂU**

Chuyên ngành : Khoa Học Máy Tính  
Mã số: 60.48.01.01

**LUẬN VĂN THẠC SĨ**

TP. HỒ CHÍ MINH, tháng 01 năm 2019

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI  
TRƯỜNG ĐẠI HỌC BÁCH KHOA –ĐHQG -HCM**

Cán bộ hướng dẫn khoa học : TS. Lê Thành Sách

Cán bộ chấm nhận xét 1 : TS. Nguyễn Hồ Mẫn Rạng

Cán bộ chấm nhận xét 2 : PGS. TS. Lê Hoàng Thái

Luận văn thạc sĩ được bảo vệ tại Trường Đại học Bách Khoa, ĐHQG  
Tp. HCM ngày 26 tháng 12 năm 2018

Thành phần Hội đồng đánh giá luận văn thạc sĩ gồm:

1. PGS. TS. Quản Thành Thơ
2. TS. Phạm Hoàng Anh
3. PGS. TS Lê Hoàng Thái
4. TS. Nguyễn Hồ Mẫn Rạng
5. TS. Nguyễn An Khương

Xác nhận của Chủ tịch Hội đồng đánh giá LV và Trưởng Khoa quản lý  
chuyên ngành sau khi luận văn đã được sửa chữa (nếu có).

**CHỦ TỊCH HỘI ĐỒNG**

**TRƯỞNG KHOA KH & KTMT**

## NHIỆM VỤ LUẬN VĂN THẠC SĨ

Họ tên học viên: NGUYỄN XUÂN THÀNH.....MSHV: 1670235.....

Ngày, tháng, năm sinh: 21/06/1992.....Nơi sinh: Bình Dương.....

Ngành: Khoa Học Máy Tính..... Mã số: 60480101.....

**I. TÊN ĐỀ TÀI:** Phát hiện và nhận dạng biển số xe sử dụng phương pháp học sâu  
(License Plate Detection and Recognition using Deep Learning).....

.....

**II. NHIỆM VỤ VÀ NỘI DUNG:** Đề tài tìm hiểu về các phương pháp phát hiện và nhận  
dạng biển số xe dựa trên học sâu. Từ đó đề xuất các cải tiến cho phương pháp cũng như  
hiện thực và đánh giá mô hình đề xuất.....

.....

**III. NGÀY GIAO NHIỆM VỤ:** 26/02/2018.....

**IV. NGÀY HOÀN THÀNH NHIỆM VỤ:** 02/12/2018.....

**V. CÁN BỘ HƯỚNG DẪN:** TS. Lê Thành Sách.....

Tp. HCM, ngày .... tháng .... năm ....

**CÁN BỘ HƯỚNG DẪN**  
(Họ tên và chữ ký)

**TRƯỞNG KHOA KH & KTMT**  
(Họ tên và chữ ký)

## LỜI CẢM ƠN

Trải qua quá trình học tập tại trường Đại học Bách Khoa thành phố Hồ Chí Minh từ lúc còn là sinh viên đại học cho tới nay đã để lại cho tôi nhiều ấn tượng sâu sắc. Tôi đã được học tập và làm việc trong một môi trường tốt, tiếp thu được nhiều kiến thức và kinh nghiệm quý báu. Tuy nhiên, có những lúc tôi phải đã trải qua nhiều khó khăn thử thách. Những lúc ấy, tôi đã nhận được sự quan tâm giúp đỡ từ gia đình, thầy cô và bạn bè. Đó cũng là động lực to lớn để tôi vượt qua những khó khăn và đạt được kết quả như ngày hôm nay.

Trước hết, tôi xin gửi lời cảm ơn sâu sắc đến gia đình, đặc biệt là ba mẹ của tôi. Ba, mẹ luôn thấu hiểu và quan tâm đến tôi cũng như là chỗ dựa tinh thần của tôi những lúc khó khăn nhất. Bên cạnh đó, ba mẹ cũng cố gắng để tạo điều kiện thuận lợi nhất cho tôi giúp tôi có thêm động lực trong học tập và làm việc.

Tiếp theo, tôi xin gửi lời cảm ơn chân thành nhất đến thầy hướng dẫn đề tài Tiến sĩ Lê Thành Sách. Thầy đã luôn theo sát, hỗ trợ cũng như định hướng cho đề tài luận văn của tôi. Bên cạnh đó tôi cũng gửi lời cảm ơn đến các thầy cô đã giảng dạy, truyền đạt lại những kiến thức và kỹ năng quý báu cho tôi.

Sau cùng, tôi xin gửi lời cảm ơn đến bạn bè và đồng nghiệp đã giúp đỡ và cho những lời khuyên hữu ích. Tôi cũng xin cảm ơn đến toàn thể nhà trường thầy cô, các bạn sinh viên đã tạo một môi trường học tập đầy năng động và hào hứng.

Một lần nữa, tôi xin chân thành cảm ơn.

Hồ Chí Minh, ngày 03 tháng 12 năm 2018  
**Nguyễn Xuân Thành**

# TÓM TẮT LUẬN VĂN

Trong thời đại cách mạng công nghiệp 4.0 hiện nay, lĩnh vực trí tuệ nhân tạo nói chung và thị giác máy tính nói riêng có vai trò đặc biệt quan trọng và tác động to lớn đến đời sống kinh tế, xã hội. Những nghiên cứu về lĩnh vực này đã được ứng dụng triển khai trong thực tế và từng bước cải thiện đời sống chúng ta ngày càng tốt hơn. Các lĩnh vực khác ngày càng được hưởng lợi từ sự phát triển của trí tuệ nhân tạo. Một trong những lĩnh vực đó là quản lý các phương tiện giao thông qua hình ảnh thu được từ camera.

Với những hình ảnh trích xuất từ camera, việc quan trọng nhất là xác định được biển số xe từ đó làm cơ sở để giải quyết các vụ việc vi phạm giao thông, quản lý việc đăng ký và sử dụng phương tiện giao thông, hỗ trợ thu phí tự động cũng như ứng dụng vào các lĩnh vực an ninh khác. Do đó việc xây dựng một ứng dụng nhận diện biển số xe thông minh có ý nghĩa và tính thực tế cao. Dưới góc độ khoa học, việc nghiên cứu các phương pháp nhận diện biển số xe mới góp phần đưa ra những cách giải quyết khác nhau cho bài toán này cũng như giải quyết những vấn đề mà các phương pháp trước chưa đạt được.

Dựa trên các nghiên cứu về học sâu, đề tài đã ứng dụng và kết hợp các phương pháp này để cho ra mô hình giúp đồng thời phát hiện và nhận dạng biển số xe. Đây là hướng nghiên cứu khá mới hiện nay và có những thuận lợi, khó khăn riêng so với phương pháp truyền thống. Những thuận lợi có thể kể tới như việc rút trích đặc trưng không còn làm một cách thủ công mà giờ đã tích hợp hoàn toàn với học máy giúp giảm đi sự phức tạp và tăng độ hiệu quả. Đặc biệt với các nghiên cứu gần đây [1] [2], việc nhận dạng biển số xe đã loại bỏ hoàn toàn công đoạn phân đoạn ký tự giúp giảm thời gian xử lý cũng như hạn chế bớt các sai sót. Hơn nữa các phương pháp học sâu phù hợp với các phần cứng tính toán song song như GPU giúp tăng tốc trong quá trình xử lý. Bên cạnh đó, vẫn tồn tại những khó khăn, thách thức và nổi bật nhất là vấn đề dữ liệu. Các phương pháp học sâu đòi hỏi một lượng dữ liệu khá lớn để có thể làm việc tốt. Dù có những hạn chế về mặt dữ liệu, bước đầu đề tài đã cho kết quả tương đối khả quan.

# **ABSTRACT**

In the industrial revolution 4.0 nowaday, Artificial Intelligence in general and Computer Vision in particular play a special role and have a great impact on economic and social life. The research in this field has been applied in practice, and gradually make our lives better. Other fields benefit from Artificial Intelligence and one of them is the vehicle management through camera.

With the images extracted from the camera, the most important is identifying the license plate which helps to resolve the traffic violations, manages the registration and using of vehicles, supports automated fees as well as applies to other security areas. Therefore, the development of a smart license plate recognition application is highly meaningful and practical. On the scientific side, the study of novel license plate recognition method contributes more solutions as well as solves problems that previous approaches have not yet achieved.

Based on the research for deep learning, the thesis propose a license plate detection and recognition method. This is a new approach which has not only some advantages but also some disadvantages. The advantages include combining the feature extraction and machine learning to reduce the complexity and increase the efficiency. In some related works [1] [2], the license plate recognition eliminates the character segmentation to reduce time processing and errors. Moreover the deep learning methods use the parallel hardware such as GPU to speed up the processing time. Beside, there are some challenges and training data is one of them. The deep learning method requires a large data to work well in real. Although there is some the data limitation, the results are positive.

## **LỜI CAM ĐOAN**

Luận văn của tôi có tham khảo các tài liệu từ nhiều nguồn khác nhau và các nguồn tham khảo này đều được trích dẫn rõ ràng trong phần tài liệu tham khảo. Ngoài những phần được trích dẫn, tôi xin cam đoan toàn bộ nội dung báo cáo là do tôi tự soạn thảo dựa trên những tìm hiểu và kết quả nghiên cứu của tôi, không sao chép từ bất kỳ tài liệu nào khác.

Tôi sẽ hoàn toàn chịu xử lý theo quy định nếu có bất kỳ sai phạm nào xảy ra liên quan đến những gì đã cam đoan.

Hồ Chí Minh, ngày 02 tháng 12 năm 2018  
**Nguyễn Xuân Thành**

# MỤC LỤC

<b>TÓM TẮT LUẬN VĂN</b>	<b>V</b>
<b>MỤC LỤC</b>	<b>VIII</b>
<b>DANH SÁCH HÌNH VẼ</b>	<b>XI</b>
<b>DANH SÁCH BẢNG</b>	<b>XIII</b>
<b>1 GIỚI THIỆU</b>	<b>1</b>
1.1 Giới thiệu đề tài . . . . .	1
1.2 Mục tiêu của đề tài . . . . .	3
1.3 Ý nghĩa của đề tài . . . . .	3
1.3.1 Ý nghĩa thực tiễn . . . . .	3
1.3.2 Ý nghĩa khoa học . . . . .	3
1.4 Phạm vi của đề tài . . . . .	4
1.5 Bố cục luận văn . . . . .	4
<b>2 CÔNG TRÌNH LIÊN QUAN</b>	<b>5</b>
2.1 Phát hiện bảng số xe . . . . .	5
2.1.1 Các phương pháp trích đặc trưng thủ công . . . . .	5
2.1.2 Các phương pháp học sâu . . . . .	6
2.2 Nhận diện bảng số xe . . . . .	9
2.2.1 Phương pháp so trùng mẫu (Template matching) . . . . .	9
2.2.2 Phương pháp học sâu . . . . .	9
2.3 Mô hình kết hợp phát hiện và nhận diện biển số xe . . . . .	12
2.3.1 Hướng tuần tự . . . . .	12
2.3.2 Hướng tích hợp . . . . .	12

<b>3 CƠ SỞ LÝ THUYẾT</b>	<b>14</b>
3.1 Mạng nơ-ron truyền thẳng . . . . .	14
3.1.1 Hàm kích hoạt . . . . .	15
3.1.2 Hàm lỗi . . . . .	17
3.1.3 Quá trình tối ưu hóa . . . . .	21
3.2 Mạng nơ-ron tích chập . . . . .	22
3.2.1 Tính chất mạng nơ-ron tích chập . . . . .	22
3.2.2 Lấy mẫu (Subsampling hay pooling) . . . . .	23
3.3 Mạng nơ-ron đệ quy . . . . .	24
<b>4 MÔ HÌNH ĐỀ XUẤT</b>	<b>28</b>
4.1 Tổng quan . . . . .	28
4.2 Phát hiện biển số xe . . . . .	29
4.3 Trích xuất vùng biển số xe . . . . .	30
4.4 Nhận dạng biển số xe . . . . .	31
<b>5 THÍ NGHIỆM</b>	<b>35</b>
5.1 Tiêu chí đánh giá . . . . .	35
5.2 Chuẩn bị dữ liệu . . . . .	36
5.2.1 Quy định về biển số xe Việt Nam . . . . .	36
5.2.2 Xây dựng tập biển số từ dữ liệu thực tế . . . . .	38
5.2.3 Sinh tập biển số . . . . .	40
5.2.4 Làm giàu dữ liệu . . . . .	41
5.3 Cài đặt thí nghiệm . . . . .	42
5.3.1 Các mô hình phát hiện biển số . . . . .	42
5.3.2 Các mô hình nhận diện biển số . . . . .	43
5.3.3 Cấu hình hệ thống thí nghiệm . . . . .	44
5.4 Kết quả thí nghiệm . . . . .	44

5.4.1	Các mô hình phát hiện biến số . . . . .	44
5.4.2	Các mô hình nhận diện biến số . . . . .	46
<b>6</b>	<b>TỔNG KẾT</b>	<b>49</b>
	<b>TÀI LIỆU THAM KHẢO</b>	<b>50</b>

# DANH SÁCH HÌNH VẼ

Hình 2-1	Các bước cơ bản của một hệ thống nhận diện biển số xe . . . . .	5
Hình 2-2	Kiến trúc mạng Faster RCNN [9] . . . . .	7
Hình 2-3	Kiến trúc mạng SSD [3] . . . . .	8
Hình 2-4	Mạng ConvNet-RNN [4] . . . . .	10
Hình 2-5	Mạng CRNN [1] . . . . .	11
Hình 2-6	Kiến trúc tổng quan của mô hình [5] . . . . .	13
Hình 3-1	Cấu trúc một nơ-ron . . . . .	14
Hình 3-2	Cấu trúc mạng truyền thẳng . . . . .	15
Hình 3-3	Hàm Sigmoid . . . . .	15
Hình 3-4	Hàm tanh . . . . .	16
Hình 3-5	Hàm ReLU . . . . .	16
Hình 3-6	So sánh giữa hàm Cross Entropy và hàm bình phương khoảng cách. Các điểm màu xanh lục thể hiện các giá trị nhỏ nhất của mỗi hàm. . . . .	18
Hình 3-7	Hàm Smooth-L1 . . . . .	19
Hình 3-8	Quá trình thu giảm chuỗi dự đoán . . . . .	20
Hình 3-9	Minh họa cách tính giá trị hàm lỗi CTC . . . . .	21
Hình 3-10	Các tính chất của mạng tích chập [6] . . . . .	23
Hình 3-11	Minh họa về lớp Max Pooling với kích thước cửa sổ 2x2 và độ dịch chuyển là 2 . . . . .	24
Hình 3-12	Mạng nơ-ron đệ quy biểu diễn dưới dạng lặp . . . . .	24
Hình 3-13	Mạng nơ-ron đệ quy biểu diễn dưới dạng không lặp . . . . .	25
Hình 3-14	Cấu trúc của mạng LSTM . . . . .	26
Hình 3-15	Mạng Bidirectional LSTM . . . . .	27
Hình 4-1	Kiến trúc tổng quan của mô hình tuần tự đề xuất . . . . .	29
Hình 4-2	Kiến trúc của mạng phát hiện biển số xe . . . . .	30

Hình 4-3	Biển số sau khi cắt ra sẽ được ghép lại và đưa về kích thước chung .	30
Hình 4-4	Kiến trúc tổng quan của mạng CRNN sử dụng trong mô hình đề xuất	31
Hình 4-5	Minh họa cho các lớp tích chập được sử dụng trong mạng CRNN .	31
Hình 4-6	Dữ liệu đầu vào sau khi qua phép biến đổi phối cảnh . . . . .	33
Hình 4-7	Kiến trúc cải tiến của mạng CRNN . . . . .	33
Hình 4-8	Minh họa cho các lớp tích chập được sử dụng trong mạng CRNN .	34
Hình 5-1	Các kích thước biển số xe ô tô . . . . .	37
Hình 5-2	Kích thước biển số xe ô tô . . . . .	38
Hình 5-3	Phân phối của từng ký tự trong tập dữ liệu . . . . .	39
Hình 5-4	Cách gán nhãn cho biển số ngắn và dài . . . . .	40
Hình 5-5	Minh họa quá trình sinh biển số ngẫu nhiên . . . . .	41
Hình 5-6	Một số mẫu biển số xe sinh tự động . . . . .	41
Hình 5-7	Một vài ảnh đã qua làm giàu . . . . .	42
Hình 5-8	Vùng đèn xe bị nhầm với biển số . . . . .	45
Hình 5-9	Kết quả phát hiện và nhận dạng biển số . . . . .	48

## **DANH SÁCH BẢNG**

Bảng 4-1	Cấu hình các lớp tích chập trong mạng CRNN[1] . . . . .	32
Bảng 5-1	Thời gian huấn luyện các mô hình trên máy . . . . .	44
Bảng 5-2	Kết quả phát hiện biển số xe . . . . .	45
Bảng 5-3	Tốc độ chạy của hai mô hình mạng phát hiện . . . . .	46
Bảng 5-4	Kết quả trên tập dữ liệu trải qua phép biến đổi . . . . .	47
Bảng 5-5	Kết quả trên tập dữ liệu thực tế và nhân tạo . . . . .	47

# DANH MỤC TỪ VIẾT TẮT

Thuật ngữ	Giải thích
CNN	Mạng nơ-ron tích chập (Convolutional Neural Network)
CRNN	Convolution Recurrent Neural Network
CTC	Connectionist Temporal Classification
FCN	Mạng nơ-ron tích chập đầy đủ (Fully Convolutional Network)
LSTM	Long Short Term Memory
RNN	Mạng nơ-ron hồi quy (Convolutional Neural Network)
SSD	Single Shot Mutilbox Detector

# CHƯƠNG 1: GIỚI THIỆU

## 1.1 Giới thiệu đề tài

Hiện nay với sự phát triển của kinh tế và xã hội, bên cạnh những mặt tích cực, nước ta vẫn còn phải đương đầu với các vấn đề phức tạp do sự phát triển gây ra. Một trong số những vấn đề đó là tình trạng giao thông quá tải và hỗn loạn thường xuyên xảy ra ở những trung tâm kinh tế lớn của cả nước. Thiệt hại từ các vấn đề giao thông gây ra là rất lớn cho nền kinh tế cũng như xã hội. Bên cạnh việc nâng cao ý thức chấp hành giao thông của người dân, việc giám sát quản lý giao thông cũng cần được đặt ưu tiên hàng đầu. Tuy nhiên, ở nước ta từ trước tới nay việc giám sát giao thông vẫn dùng phương pháp thủ công là chủ yếu. Việc giám sát thủ công nhìn chung rất khó khăn vì số lượng phương tiện giao thông lớn, đòi hỏi phải có cách tiếp cận một cách tự động để giảm nhẹ sức lao động của con người. Đây là hướng đi đã có từ lâu ở những nước phát triển. Hiện nay ở nước ta trước yêu cầu năm bắt công nghệ trong thời đại cách mạng công nghiệp 4.0 để ứng dụng cho đời sống sản xuất, việc giải quyết bài toán giám sát giao thông tự động là vô cùng cấp thiết. Bài toán giám sát giao thông tự động bao gồm nhiều bài toán con, một trong số đó là bài toán nhận diện biển số xe giúp quản lý phương tiện giao thông. Với các lý do nêu trên, đề tài hình thành với mục tiêu giải quyết tốt bài toán nhận diện biển số xe để góp phần giải quyết bài toán giao thông chung hiện nay.

Vấn đề phát hiện và nhận diện biển số xe là một trong những hướng nghiên cứu đã có từ lâu trong lĩnh vực thị giác máy tính. Tuy nhiên, hiện nay với sự phát triển mạnh mẽ của phương pháp học sâu (deep learning) đã đem lại hướng tiếp cận mới cho vấn đề này. Có thể kể tới các mạng nơ-ron trong học sâu phổ biến hiện nay như Convolution Neural Networks (CNN), Recurrent Neural Networks (RNN) cùng các kiến trúc mạng như LeNet[7], ImageNet [8], Fast R-CNN [9], .... Đây là hướng nghiên cứu khá mới nhưng lại đạt được những kết quả hết sức ấn tượng qua các cuộc thi trong giới học thuật cũng như những ứng dụng thực tế.

Bên cạnh những điểm tích cực nêu trên, vấn đề nhận diện biển số xe cũng có một số khó khăn thách thức bao gồm các yếu tố do môi trường, các yếu tố do quá trình thu nhận hình ảnh và các yếu tố liên quan tới quá trình huấn luyện mạng học sâu:

- Các yếu tố do môi trường có thể kể đến như độ phức tạp của khung cảnh có chứa biển số xe, gây khó khăn cho việc phân biệt biển số xe với các đối tượng khác. Ngoài ra vấn đề độ sáng của môi trường, độ nhạy sáng của các thiết bị cảm biến cũng ảnh hưởng rất lớn đến kết quả của quá trình phát hiện và nhận dạng.
- Các yếu tố do quá trình thu nhận hình ảnh gồm có độ mờ, chất lượng ảnh thấp và độ biến dạng của ảnh. Vấn đề độ mờ xuất phát chủ yếu từ việc căn chỉnh tiêu cự của máy quay và sự chuyển động của máy quay hoặc vật thể gây ra. Hình ảnh hay video trải qua quá trình nén và giải nén cũng sẽ dẫn đến giảm chất lượng hình ảnh. Cuối cùng với nhiều góc độ của máy quay, hình ảnh thu được sẽ bị biến dạng.
- Quá trình huấn luyện mạng deep learning cần nhiều dữ liệu để đảm bảo được tính tổng quát cao cũng như độ chính xác tốt. Tuy nhiên việc thu thập và gán nhãn số lượng lớn dữ liệu là công việc mất rất nhiều thời gian và công sức.
- Việc huấn luyện cũng tốn nhiều thời gian. Kích thước mạng càng lớn và phức tạp thì thời gian cũng tăng theo. Hơn nữa để tìm được bộ siêu tham số phù hợp với kiến trúc mạng, với mỗi một tham số truyền vào ta phải thực hiện việc huấn luyện lại mạng.

Ngoài ra, qua khảo sát các đề tài hiện nay, vẫn còn một số vấn đề còn tồn tại sau:

- Các phương pháp nhận diện ký tự ứng dụng học sâu có một vài giới hạn trong việc phát hiện các ký tự có góc xoay bất kỳ. Đa phần các phương pháp đó chỉ áp dụng tốt với các góc xoay không quá lớn.
- Đa phần các phương pháp sử dụng học sâu chưa được chứng minh sử dụng được với biển số xe ở Việt Nam. Các dạng biển số xe mà các phương pháp đó chạy được có kích thước tương đối đồng nhất và cách bố trí các ký tự không quá phức tạp như biển số xe ở Việt Nam.

- Tốc độ xử lý của các giải thuật hiện nay chưa đạt đến thời gian thực.

Với những khó khăn kể trên, việc cải tiến, kết hợp các phương pháp hiện tại để cho ra đời một phương pháp hiệu quả giúp giải quyết các vấn đề trên là hoàn toàn cấp thiết. Bên cạnh đó, đề tài cũng mong muốn có thể giải quyết được các vấn đề còn tồn tại của các đề tài đi trước.

## 1.2 Mục tiêu của đề tài

Mục tiêu của đề tài là đề xuất một phương pháp phát hiện và nhận dạng biển số xe Việt Nam hiệu quả. Phương pháp chủ yếu được sử dụng là học sâu. Yêu cầu cần đạt được bao gồm độ chính xác cho việc phát hiện biển số là trên 90% và độ chính xác cho việc nhận diện chuỗi biển số là trên 80%. Ngoài ra đề tài cũng chú trọng đến thời gian thực thi của mô hình đề xuất.

## 1.3 Ý nghĩa của đề tài

### 1.3.1 Ý nghĩa thực tiễn

Phương pháp mà đề tài đề xuất có thể được sử dụng trong các ứng dụng như giám sát giao thông tự động, bãi giữ xe thông minh, trạm thu phí tự động ... nhằm giảm bớt thời gian, công sức của con người, giảm tình trạng kẹt xe và tăng sự tiện lợi cho người tham gia giao thông. Xây dựng được tập dữ liệu chính xác, đa dạng và sát với điều kiện thực tế nước ta.

### 1.3.2 Ý nghĩa khoa học

Đóng góp một phương pháp phát hiện và nhận diện biển số xe hiệu quả, tăng thêm độ chính xác cũng như giải quyết được các vấn đề còn tồn tại của các phương pháp đi trước. Ngoài ra đề tài sẽ làm cơ sở cho các nghiên cứu trong nước về sau.

## **1.4 Phạm vi của đề tài**

Biển số xe xuất hiện trong tập dữ liệu theo phương ngang, không bị che khuất, độ mờ và độ biến dạng thấp. Vị trí của camera dùng để thu thập hình ảnh là cố định hoặc di động với góc lệch không quá lớn. Hình ảnh thu được có thể nhiều từ thời tiết như mưa, sương mù, . . . Tuy nhiên các hình ảnh này có thể nhận dạng được đầy đủ các ký tự bằng mắt thường được. Biển số xe dùng trong đề tài là biển số xe máy và ô tô ở Việt Nam. Biển số này có kích thước, ký hiệu và bố trí tuân thủ theo Thông tư 15/2014/TT-BCA. Các biển số xe của các nước khác không thuộc phạm vi đề tài. Kết quả của mô hình là vị trí của biển số xe cũng như chuỗi ký tự trong biển số đó.

## **1.5 Bố cục luận văn**

Trong báo cáo này có tất cả 7 chương. Chương 2 có mục đích giới thiệu các công trình nghiên cứu liên quan, các hướng giải quyết cho bài toán phát hiện và nhận dạng. Chương 3 cung cấp một số cơ sở lý thuyết đặt nền móng cho các mô hình được đề xuất trong chương 4. Chương 5 đưa ra thông tin thí nghiệm, các chỉ tiêu đánh giá và kết quả đánh giá của các phương pháp cùng với một số nhận xét dựa vào các kết quả đấy. Chương 6 kết luận báo cáo và trình bày hướng phát triển trong tương lai. Chương 7 trình bày danh sách các tài liệu tham khảo được trích dẫn trong đề tài.

# CHƯƠNG 2: CÔNG TRÌNH LIÊN QUAN

Bài toán nhận dạng biển số xe là một bài toán con của bài toán nhận diện văn bản ngoại cảnh. Do đó các công trình nghiên cứu liên quan của hai bài toán này thường có mối liên hệ chặt chẽ với nhau. Đây là bài toán đã và đang được các nhóm nghiên cứu quan tâm và có sự phát triển mạnh mẽ trong những năm gần đây. Thông thường bài toán nhận diện biển số xe bao gồm các bước sau:



**Hình 2-1:** Các bước cơ bản của một hệ thống nhận diện biển số xe

Từ các bước trên ta nhận thấy có ba bài toán chính cần giải quyết ở đây. Đó là bài toán phát hiện vị trí biển số trong hình, bài toán phân đoạn ký tự và bài toán nhận diện chuỗi ký tự trong biển số. Dựa trên những công trình nổi bật gần đây, đề tài nhận thấy hướng nhận dạng chuỗi biển số không qua bước phân đoạn ký tự rất tiềm năng vì có khá nhiều ưu điểm. Do đó đề tài tập trung vào khảo sát các công trình liên quan đến việc phát hiện vùng biển số và nhận diện ký tự. Ngoài ra trong số các công trình nghiên cứu gần đây còn đề cập đến vấn đề kết hợp xử lý cả hai quá trình phát hiện và nhận dạng một cách đồng thời.

## 2.1 Phát hiện bảng số xe

### 2.1.1 Các phương pháp trích đặc trưng thủ công

Các phương pháp trích đặc trưng thủ công được sử dụng nhiều trong các giai đoạn trước đây. Các đặc trưng ở đây thường là các đặc trưng cơ bản như góc, cạnh, vân ảnh, màu sắc, mức sáng. Để trích xuất các đặc trưng này, các phương pháp phổ biến được dùng đến như nhị phân hóa ảnh, SIFT, HOG, phân tích thành phần liên thông và hình thái học. Sau đó các đặc trưng này được đưa qua các bộ phân loại như AdaBoost, SVM.

Ưu điểm: các phương pháp trích xuất đặc trưng cơ bản nhìn chung đơn giản, dễ hiện

thực và nhanh. Trong một số trường hợp, khi kết hợp nhiều đặc trưng lại với nhau cho ra độ chính xác tương đối cao.

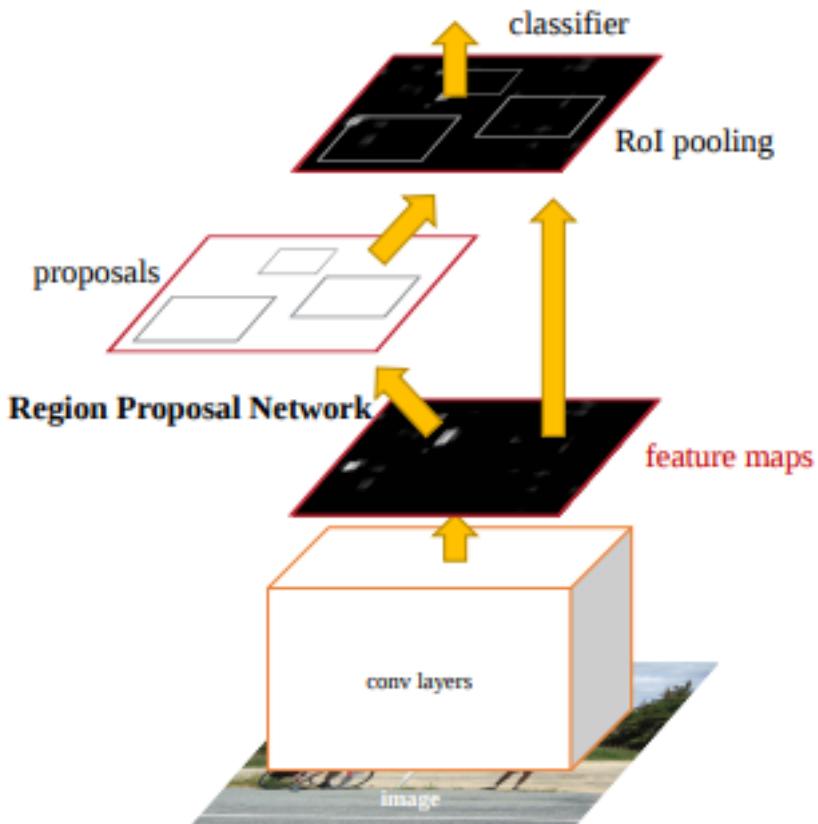
Nhược điểm: các phương pháp này khá nhạy cảm với sự thay đổi mức sáng, nhiễu và mờ. Hơn nữa việc thiết kế bộ rút trích đặc trưng cũng phụ thuộc khá nhiều vào tập dữ liệu.

### 2.1.2 Các phương pháp học sâu

Ngày nay các phương pháp để phát hiện đối tượng sử dụng phương pháp học sâu rất đa dạng. Có thể kể tới như các kiến trúc mạng như Faster RCNN [9], SSD [3], YOLO [11] và YOLO2 [11]. Các kiến trúc này tỏ ra hiệu quả trong việc phát hiện nhiều đối tượng trong cùng một ảnh.

#### Mạng Faster RCNN [9]

Mạng Faster RCNN là một mạng cải tiến từ mạng Fast RCNN. Cấu trúc mạng này gồm 3 phần: phần mạng rút trích đặc trưng cơ bản, lớp mạng đề xuất vùng dự tuyển Region Proposal Network (RPN) và cuối cùng là lớp mạng dự đoán vị trí cũng như phân loại đối tượng. Ảnh đầu vào sau khi qua lớp mạng rút trích đặc trưng sẽ cho ra bản đồ đặc trưng (feature map). Sau đó bản đồ đặc trưng này sẽ được cho qua mạng RPN để cho ra các vùng dự tuyển. Các vùng dự tuyển sẽ được đưa về kích thước cho trước bằng lớp Roi Pooling. Việc phân loại đối tượng cũng như dự đoán vị trí sẽ được thực hiện trên các vùng dự tuyển đó. Cải tiến lớn nhất của Faster RCNN đến từ việc sử dụng lớp mạng RPN thay cho giải thuật tìm kiếm vét cạn. Từ đó tốc độ chạy của mô hình được cải thiện đáng kể. Hình 2-2 minh họa cho kiến trúc mạng Faster RCNN.



**Hình 2-2:** Kiến trúc mạng Faster RCNN [9]

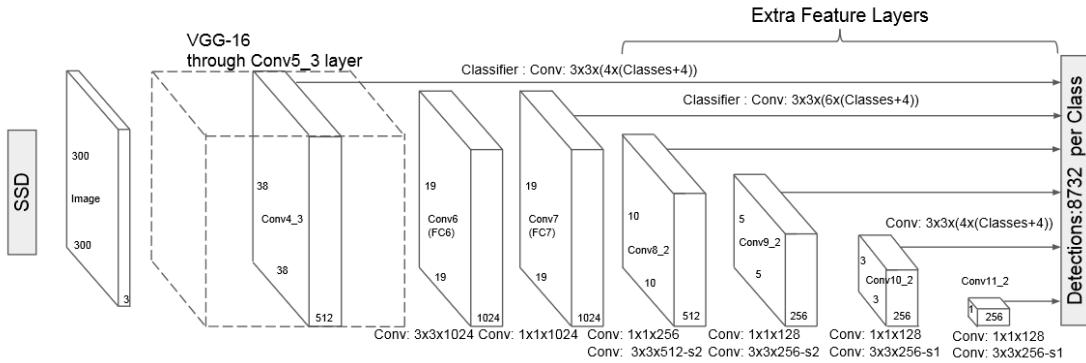
Ưu điểm: tốc độ cũng như độ chính xác được cải thiện hơn so với phiên bản trước đó.

Nhược điểm: việc sử dụng một lớp bản đồ đặc trưng có thể bỏ sót các đối tượng nhỏ từ đó dẫn tới độ chính xác thấp hơn khi so với các phương pháp học sâu khác.

### Mạng Single Shot Multibox Detector [3]

SSD là một mạng học sâu giúp phát hiện và phân loại đối tượng. Khác với phương pháp phát hiện bằng cửa sổ trượt (sliding window), thay vì sử dụng một (một số) cửa sổ có kích thước cố định, thì SSD sinh ra một số lượng hữu hạn các ô chuẩn (default box) để rồi từ các ô chuẩn đó, hệ thống có thể xác định vị trí cũng như lớp của các đối tượng trong quá trình huấn luyện. Việc đưa vào các ô chuẩn giúp mạng có thể dễ dàng học cách dự đoán cả kích thước của các đường bao chữ nhật quanh vị trí của đối tượng. Các ô chuẩn này được áp dụng trên nhiều lớp bản đồ đặc trưng đầu ra với các kích thước khác nhau. Điều đó giúp SSD có thể phát hiện được những đối tượng có kích thước lớn nhỏ

khác nhau. Ngoài ra mạng SSD còn là một mạng nơ-ron tích chập đầy đủ từ lớp mạng nền cho tới các lớp dự đoán vị trí và phân loại. Mạng SSD gồm các thành phần cơ bản như bộ làm giàu dữ liệu đầu vào, lớp mạng nền trích đặc trưng, lớp mạng phát hiện và phân loại. Hình 2-3 mô tả kiến trúc tổng quát của mạng SSD.



**Hình 2-3: Kiến trúc mạng SSD [3]**

Ưu điểm: phương pháp này tích hợp việc phân loại và phát hiện vị trí trong cùng một mạng. Ngoài ra độ chính xác của phương pháp cũng như thời gian xử lý được cải thiện hơn so với phương pháp cửa sổ trượt.

Nhược điểm: số lượng các ô chuẩn lớn sẽ gây khó khăn cho quá trình huấn luyện do đòi hỏi nhiều bộ nhớ hơn.

### Mạng YOLOv2 [11]

Mạng YOLOv2 là cải tiến của mạng YOLO. Mục tiêu mà YOLOv2 hướng tới đó là phát hiện đối tượng trong thời gian thực. Do đó thiết kế mạng của YOLOv2 có phần gọn nhẹ hơn so với Faster RCNN hay SSD. YOLOv2 kế thừa cơ chế hoạt động của YOLO, trong đó ảnh đầu vào sẽ được chia lưới và ứng với mỗi ô trong lưới đó sẽ được dùng để dự đoán vị trí của những đường bao cũng như độ tin cậy. Khác với Faster RCNN và SSD sử dụng hàm lỗi Cross Entropy để phân loại đối tượng, YOLOv2 sử dụng hàm Mean Square Error (MSE). Các cải tiến trong YOLOv2 giúp tăng độ chính xác cho việc phát hiện và phân loại đối tượng bao gồm việc huấn luyện trên những mức độ co giãn ảnh đầu vào khác nhau (Multi-scale training) và kích thước ô chuẩn. Trong quá trình huấn luyện kích

thước ảnh đầu vào sẽ được chọn ngẫu nhiên mỗi 10 bộ ảnh trong khoảng kích thước từ  $\{320, 352, \dots, 608\}$ . Về kích thước các ô chuẩn YOLOv2 sử dụng giải thuật k-mean để gom cụm dựa trên tập huấn luyện. Với các cải tiến trên, mạng YOLOv2 có thể nhận dạng được các đối tượng có kích thước khác nhau tốt hơn, đặc biệt là đối tượng nhỏ.

Ưu điểm: tốc độ chạy cao rất phù hợp với các ứng dụng thời gian thực. Mô hình YOLOv2 cũng chiếm ít bộ nhớ hơn so với các mạng khác.

Nhược điểm: độ chính xác nhìn chung thấp hơn so với các phương pháp khác. Tuy nhiên cũng tùy vào yêu cầu ứng dụng mà ta có thể lựa chọn cấu hình phù hợp để cân bằng giữa tốc độ chạy và độ chính xác.

## 2.2 Nhận diện biển số xe

### 2.2.1 Phương pháp so trùng mẫu (Template matching)

So trùng mẫu là phương pháp nhằm phân loại đối tượng dựa trên sự tương tự giữa những đối tượng đang xem xét với mẫu cho trước. Với bài toán nhận diện biển số xe mẫu ở đây là các ký tự. Phương pháp này sử dụng chủ yếu trên ảnh xám hay ảnh nhị phân. Tiêu biểu cho phương pháp này là các công trình [12], [13]. Việc so trùng sử dụng các độ đo khoảng cách như khoảng cách Mahalanobis, khoảng cách Jaccard như trong [13] và khoảng cách Hamming. Với phương pháp này, độ chính xác mà công trình [13] đạt được là 97.2%.

Ưu điểm: phương pháp này đơn giản và dễ hiện thực.

Nhược điểm: Trong thực tế phương pháp này khó áp dụng vì đòi hỏi kích thước của các ký tự phải cố định. Bên cạnh đó phương pháp này cũng khá nhạy cảm với nhiễu, sự thay đổi mức sáng cũng như góc quay.

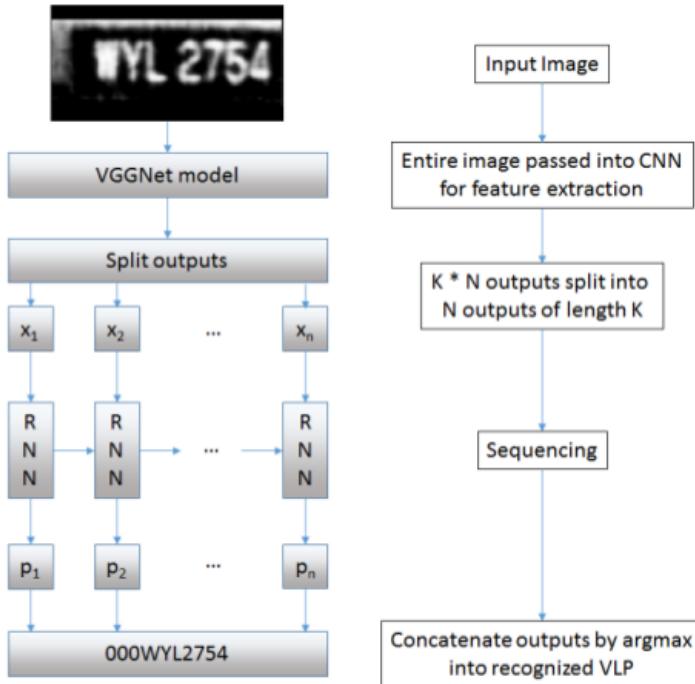
### 2.2.2 Phương pháp học sâu

Các phương pháp nhận diện biển số xe đa phần hoạt động trên tập dữ liệu là hình ảnh biển số xe được cắt sẵn. Các ảnh cắt sẵn này chỉ được gán nhãn chuỗi ký tự mà không

gán nhãn vị trí của từng ký tự trong hình. Ngoài ra các chuỗi ký tự thường chỉ nằm trên một hàng.

### Mạng ConvNet-RNN [4]:

Kiến trúc mạng được sử dụng bao gồm: lớp mạng VGG [14] và lớp mạng RNN. Đầu tiên ảnh đầu vào được cho chạy qua lớp mạng VGG [14] để lấy đặc trưng. Sau đó các đặc trưng được tách thành vector và đưa vào lớp kế tiếp là RNN. Tại lớp này, các vector sẽ được gán nhãn thành các ký tự tương ứng. Hình 2-4 mô tả cho phương pháp này.



**Hình 2-4: Mạng ConvNet-RNN [4]**

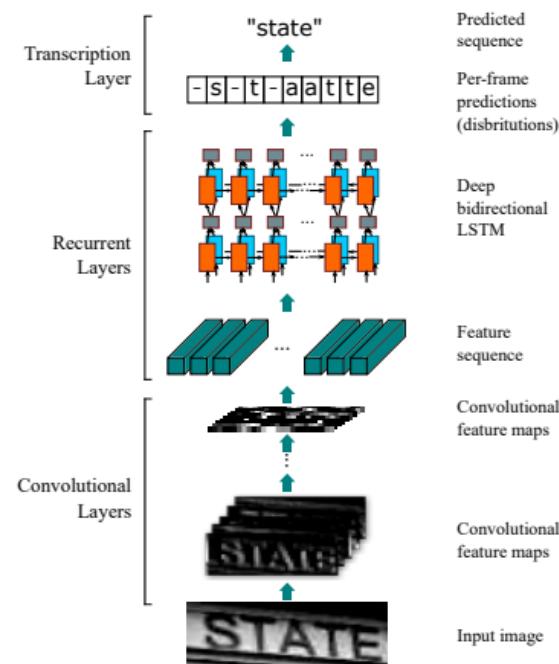
**Ưu điểm:** phương pháp này có hiệu quả tốt hơn so với phương pháp cửa sổ trượt. Ngoài ra dữ liệu huấn luyện đầu vào cũng không cần được phân đoạn trước.

**Nhược điểm:** phương pháp này có hạn chế là tỷ lệ nhận dạng sai khá cao cho các cặp ký tự M và N, D và Q, T và Y, C và G.

### Mạng CRNN [1]:

Một công trình khác cũng liên quan tới vấn đề nhận diện bảng số xe đó là phương pháp của Shi [1] và đồng sự. Trong phương pháp này, nhóm tác giả sử dụng 2 lớp mạng chính: lớp mạng CNN để trích xuất đặc trưng và lớp mạng Long-Short Term Memory

(LSTM) [15] cho việc nhận dạng các ký tự. Lớp mạng LSTM gồm 2 khối LSTM chạy độc lập với nhau theo 2 chiều hay còn được gọi là Bidirectional LSTM (BiLSTM). Bản đồ đặc trưng có được sau khi qua lớp mạng CNN sẽ được biến đổi sao cho phù hợp với đầu vào vào của lớp mạng BiLSTM. Cuối cùng lớp BiLSTM được liên kết với hàm lối Connectionist Temporal Classification (CTC) [16] cho phép chuyển đổi các vector đặc trưng thành chuỗi các xác suất và tìm ra được chuỗi có tổng xác suất là lớn nhất. Hàm lối CTC tương tự như giải thuật Hidden Markop Model (HVM) tuy nhiên lại có độ hiệu quả cao hơn so với HVM hoặc HVM kết hợp học sâu. Hình 2-5 mô tả về cấu trúc của mạng CRNN.



**Hình 2-5: Mạng CRNN [1]**

**Ưu điểm:** phương pháp này có tính tổng quát tương đối cao. Không chỉ áp dụng cho biển số xe mà có thể áp dụng cho các văn bản ngoại cảnh, bản nhạc . . . . Hơn nữa phương pháp này chỉ dùng tập dữ liệu sinh tự động để huấn luyện nhưng vẫn đạt được kết quả tốt trên các tập dữ liệu thực khác.

**Nhược điểm:** phương pháp chỉ nhận dạng được chuỗi dữ liệu trên cùng một hàng.

## 2.3 Mô hình kết hợp phát hiện và nhận diện biển số xe

### 2.3.1 Hướng tuần tự

Hướng tuần tự có sự tách biệt giữa quá trình phát hiện và quá trình nhận diện. Quá trình nhận diện có thể phản hồi thông tin về cho quá trình phát hiện.

#### **Phương pháp của Masood và đồng sự [17]:**

Phương pháp này áp dụng 3 mạng CNN để phát hiện và nhận dạng biển số xe. Mạng đầu tiên để phát hiện và phân loại biển số xe, mạng thứ 2 để phát hiện các ký tự và mạng cuối cùng để nhận dạng ký tự.

Ưu điểm: đạt được kết quả khá cao trên các tập dữ liệu biển số xe của Mỹ và châu Âu với độ chính xác lần lượt là 93.44% và 94.55%.

Nhược điểm: mô hình mà phương pháp này đề xuất khá lớn gồm 3 mạng tách biệt nên sẽ tốn thời gian huấn luyện.

#### **Phương pháp của Li Hui và đồng sự [18]:**

Phương pháp này sử dụng một mạng phân loại CNN để tạo ra bản đồ xác suất của những pixel là ký tự. Bản đồ này sau đó sẽ được gom nhóm lại nhờ giải thuật Non-Maxima Supression (NMS) và Run Length Smoothing (RLSA). Từ các nhóm pixel này các đường bao sẽ được tạo ra nhờ vào giải thuật phân tích các thành phần liên kết (CCA). Sau đó, từ các nhóm pixel này vùng biển số sẽ được trích xuất và đưa qua mô hình tương tự như [1] để nhận diện chuỗi biển số.

Ưu điểm: phương pháp đạt độ chính xác khá cao so với các phương pháp trước đó (97.56%)

Nhược điểm: bước trích xuất biển số xe vẫn còn theo cách thủ công dễ gây ra sai số cho bước nhận dạng phía sau.

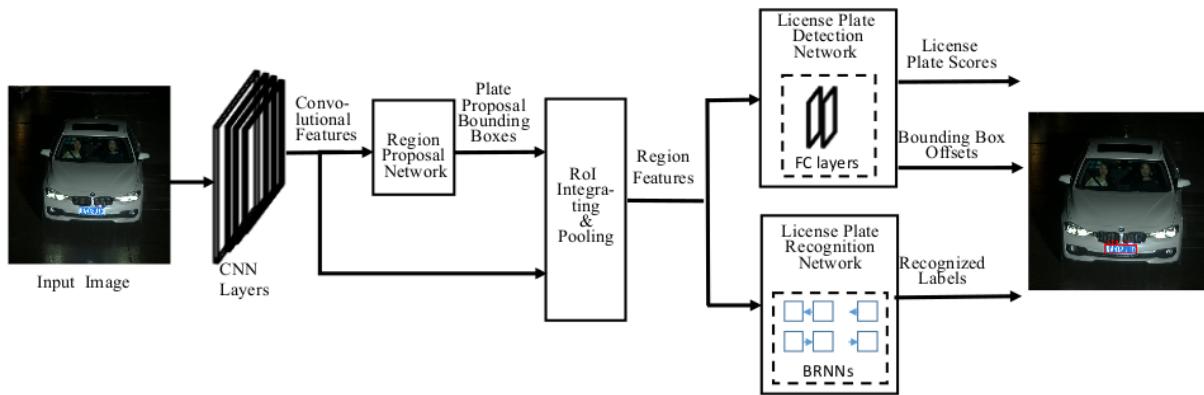
### 2.3.2 Hướng tích hợp

Hướng tích hợp là hướng mà việc kết hợp giữa quá trình phát hiện và quá trình nhận diện. Giữa quá trình phát hiện và nhận dạng sẽ chia sẻ thông tin thông qua bộ phân loại

trung gian. Một số phương pháp có sử dụng bước tiền xử lý để xác định vùng quan tâm (ROI).

### **Phương pháp của Li Hui và đồng sự [5]:**

Phương pháp này lấy ý tưởng kết hợp các mạng có sẵn thành một mạng duy nhất cho việc phát hiện và nhận dạng biển số xe. Kiến trúc mạng bao gồm các phần: lớp mạng trích xuất đặc trưng cấp thấp, tạo vùng chứa biển số, xử lý vùng chứa biển số, phát hiện và nhận dạng biển số. Hình 2-6 minh họa về tổng quan kiến trúc mạng.



**Hình 2-6: Kiến trúc tổng quan của mô hình [5]**

**Ưu điểm:** phương pháp này làm giảm các bước trung gian, đặc biệt là bước phân đoạn ký tự. Phương pháp này cũng hoạt động tốt trong các điều kiện tự nhiên như là ban ngày và ban đêm, mưa và nắng, ....

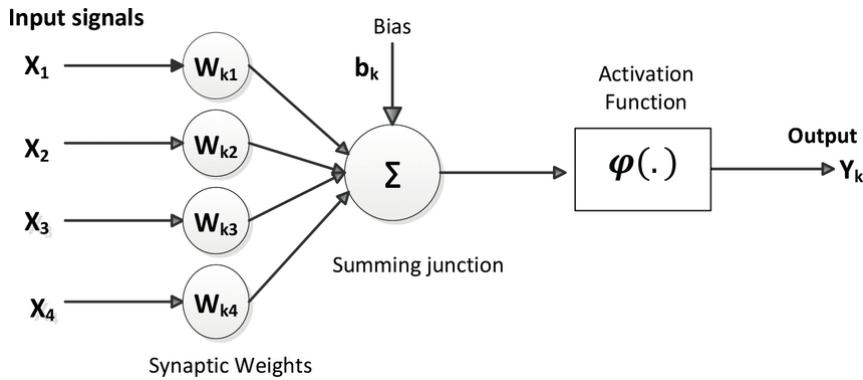
**Nhược điểm:** quá trình nhận dạng ký tự chỉ hoạt động với chuỗi ký tự trên một hàng. Chiến thuật huấn luyện tương đối phức tạp và cần một lượng lớn dữ liệu.

# CHƯƠNG 3: CƠ SỞ LÝ THUYẾT

Chương này trình bày về một số lý thuyết nền tảng. Các kiến thức này tập trung chủ yếu về phương pháp học sâu. Đây là cơ sở để xây dựng nên các phương pháp đề xuất.

## 3.1 Mạng nơ-ron truyền thẳng

Các kiến trúc mạng học sâu ngày nay đều bắt nguồn từ mạng nơ-ron truyền thẳng. Với dữ liệu đầu vào là một vector  $x_i$ , nhãn của dữ liệu đó là  $y_i$ , thì một mạng nơ-ron truyền thẳng có thể được xem như một hàm số phi tuyến tính  $f_W(x)$  với  $W$  là các hệ số giúp ánh xạ dữ liệu đầu vào  $x_i$  thành nhãn tương ứng  $y_i$ . Một mạng nơ-ron truyền thẳng có cấu tạo từ nhiều đơn vị nơ-ron nối tiếp nhau. Hình 3-1 mô tả cấu trúc tổng quát của một đơn vị mạng nơ-ron.

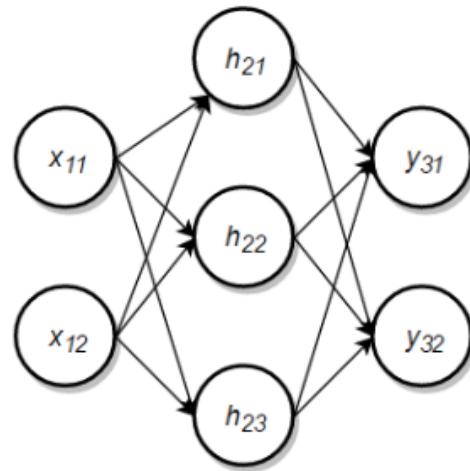


Hình 3-1: Cấu trúc một nơ-ron

Một đơn vị này bao gồm vector dữ liệu đầu vào  $x_i$ , ma trận trọng số (weights)  $w_{ij}$ , hệ số bias  $b_i$ , hàm tính tổng trọng số và hàm kích hoạt. Hàm tính tổng trọng số được định nghĩa như công thức sau:

$$o_j = \sum_{i=1}^n w_{ji}x_i + b_j \quad (3.1)$$

Sau đó  $o_j$  sẽ được đưa qua hàm kích hoạt phi tuyến để có được giá trị đầu ra sau cùng. Việc tính toán sẽ tiếp tục nếu vẫn còn những lớp mạng phía sau. Một mạng nơ-ron truyền thẳng thông thường được tổ chức như hình 3-2 bao gồm: một lớp đầu vào, một hay nhiều lớp ẩn (hidden), và một lớp đầu ra.



Lớp đầu vào      Lớp ẩn      Lớp đầu ra

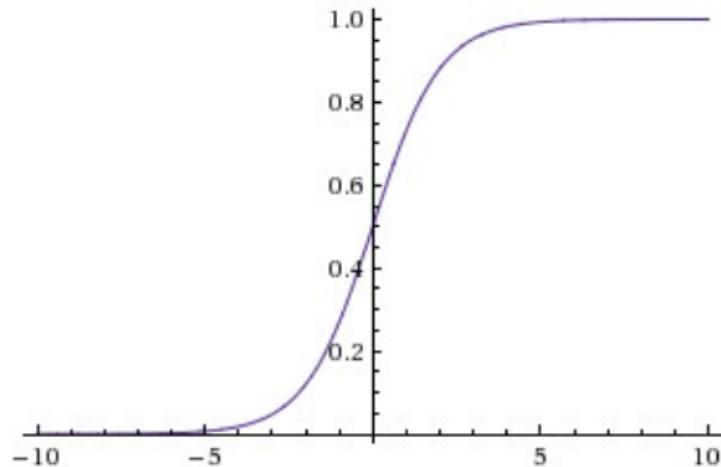
**Hình 3-2:** Cấu trúc mạng truyền thẳng

### 3.1.1 Hàm kích hoạt

Như đã đề cập ở trên, một thành phần không thể thiếu trong mạng nơ-ron đó là hàm kích hoạt (activation function). Hàm kích hoạt giúp tạo ra các thông tin trùu tượng cho các lớp phía sau. Một số hàm kích hoạt thường sử dụng như:

#### Sigmoid

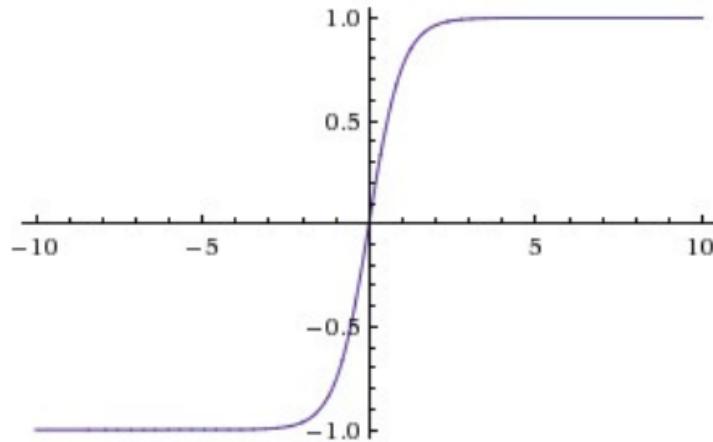
$$f(x) = \frac{1}{1 + e^{-x}}$$



**Hình 3-3:** Hàm Sigmoid

## Tanh

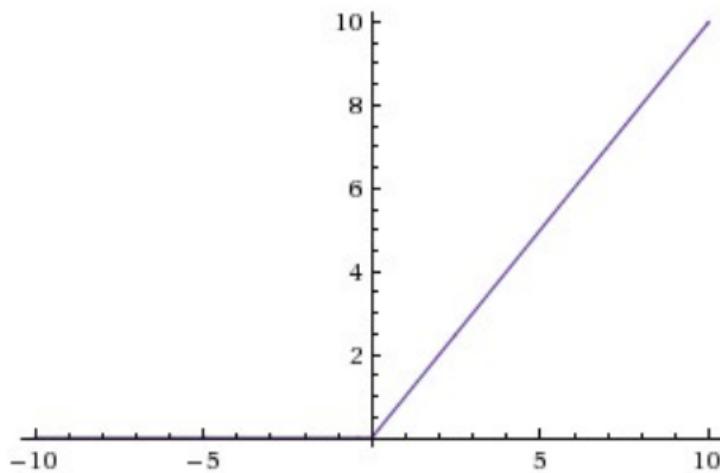
$$f(x) = \tanh(x)$$



Hình 3-4: Hàm tanh

## ReLU

$$f(x) = \max(0, x)$$



Hình 3-5: Hàm ReLU

## Leaky ReLU

$$f(x) = \max(0.1 * x, x)$$

## Maxout

$$f(x) = \max(\omega_1^T x + b_1, \omega_2^T x + b_2)$$

## ELU

$$f(n) = \begin{cases} x & \text{if } x < 0 \\ \alpha(\exp(x) - 1) & \text{if } x \leq 0 \end{cases}$$

Trong thực tế, hàm kích hoạt ReLU và Leaky Relu là hai hàm thường được dùng nhất. Hàm Tanh cũng được sử dụng hạn chế còn hàm Sigmoid gần như ít được sử dụng.

### 3.1.2 Hàm lỗi

Trong quá trình huấn luyện mạng nơ-ron, mục tiêu là giảm thiểu sai số giữa dự đoán và nhãn. Từ đó việc xây dựng hàm lỗi đóng một vai trò rất quan trọng. Hàm lỗi có hai tính chất quan trọng là trả về giá trị không âm và phải có đạo hàm liên tục. Theo cách hiểu đơn giản hàm lỗi sẽ phạt mô hình mỗi khi nó dự đoán sai và tỷ lệ thuận với mức độ sai của dự đoán. Có rất nhiều hàm lỗi được sử dụng chỉ những bài toán khác nhau, tuy nhiên đề tài chỉ tập trung một số hàm lỗi thông thường được sử dụng trong các bài toán phát hiện và nhận dạng như hàm Mean Square Error (MSE), Cross Entropy, Smooth-L1 và CTC.

#### Hàm lỗi Mean Square Error (MSE):

Đây là hàm lỗi đơn giản đáp ứng được các tính chất đã đề ra và cũng dễ sử dụng nhất. Hàm này định nghĩa bởi công thức:

$$E(W) = \frac{1}{N} \sum_{i=1}^N ||t_i - y_i||^2 \quad (3.2)$$

Hàm này thường được ứng dụng trong các bài toán đơn giản như hồi quy tuyến tính hay phân loại tập số liệu. Tuy nhiên hàm số này khó có thể áp dụng cho các bài toán phức tạp hơn.

#### Hàm lỗi Cross Entropy:

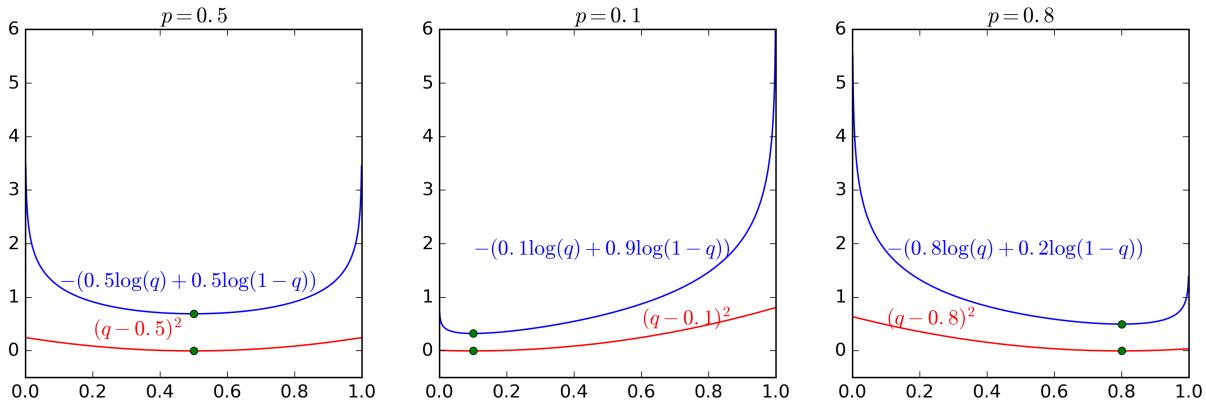
Hàm này hay được sử dụng trong các bài toán phân loại. Cross Entropy giữa hai phân phối  $p$  và  $q$  được định nghĩa là:

$$H(p, q) = E_p[-\log q] \quad (3.3)$$

Với  $p$  và  $q$  rời rạc công thức trên được viết lại:

$$H(p, q) = - \sum_{i=1}^C p_i \log q_i \quad (3.4)$$

Hàm Cross Entropy có nhiều ưu điểm so với hàm bình phương khoảng cách thường. Hình 3-6 minh họa cho hàm lỗi trong trường hợp  $C = 2$  và  $p_1$  lần lượt có giá trị là 0.5, 0.1 và 0.8.



**Hình 3-6:** So sánh giữa hàm Cross Entropy và hàm bình phương khoảng cách. Các điểm màu xanh lục thể hiện các giá trị nhỏ nhất của mỗi hàm.

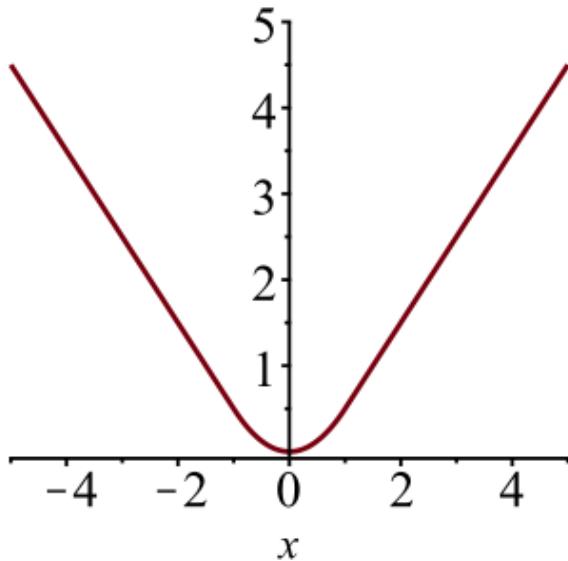
Như hình 3-6 ta nhận thấy hàm Cross Entropy nhận giá trị rất cao khi  $p$  ở xa  $q$ . Trong khi đó với hàm bình phương khoảng cách sự chênh lệch là không đáng kể. Do đó mô hình được huấn luyện với hàm Cross Entropy sẽ cho kết quả tốt hơn vì với mỗi dự đoán sai sẽ bị trừng phạt nặng hơn.

### Hàm lỗi Smooth-L1

Hàm lỗi này được dùng trong các mạng phát hiện vật thể với đường bao. Hàm này giúp hồi quy các đường bao dự đoán về các đường bao đúng. Hàm có công thức như sau:

$$\begin{cases} 0.5x^2 & x < 1 \\ -(n+1)/2 & \text{otherwise} \end{cases}$$

Hình minh họa về hàm Smooth L1:



**Hình 3-7:** Hàm Smooth-L1

### Hàm lỗi CTC

Hàm lỗi Connectionist Temporal Classification [16] (CTC) được sử dụng trong các bài toán nhận dạng chuỗi ký tự. Đây là hàm lỗi được sử dụng trong kiến trúc mạng CRNN [1] cho việc nhận dạng văn bản. Ưu điểm của hàm lỗi này là có thể nhận diện chuỗi ký tự có chiều dài bất kỳ. Một vài khái niệm quan trọng trong hàm lỗi này như:

Với mỗi chuỗi đầu vào, sẽ có nhiều chuỗi dự đoán có chiều dài khác nhau tương ứng với chuỗi ban đầu. Tuy nhiên những chuỗi dự đoán này đều có thể thu giảm về chuỗi ban đầu. Những ký tự trong chuỗi ban đầu có thể được lặp lại nhiều lần. Ở giữa các ký tự hợp lệ là các ký tự trống  $\epsilon$ . Các ký tự trống này cùng với các ký tự lặp lại sẽ bị loại bỏ trong quá trình thu giảm.

h h e ε ε | | | ε | | o

h e ε | ε | o

h e | | o

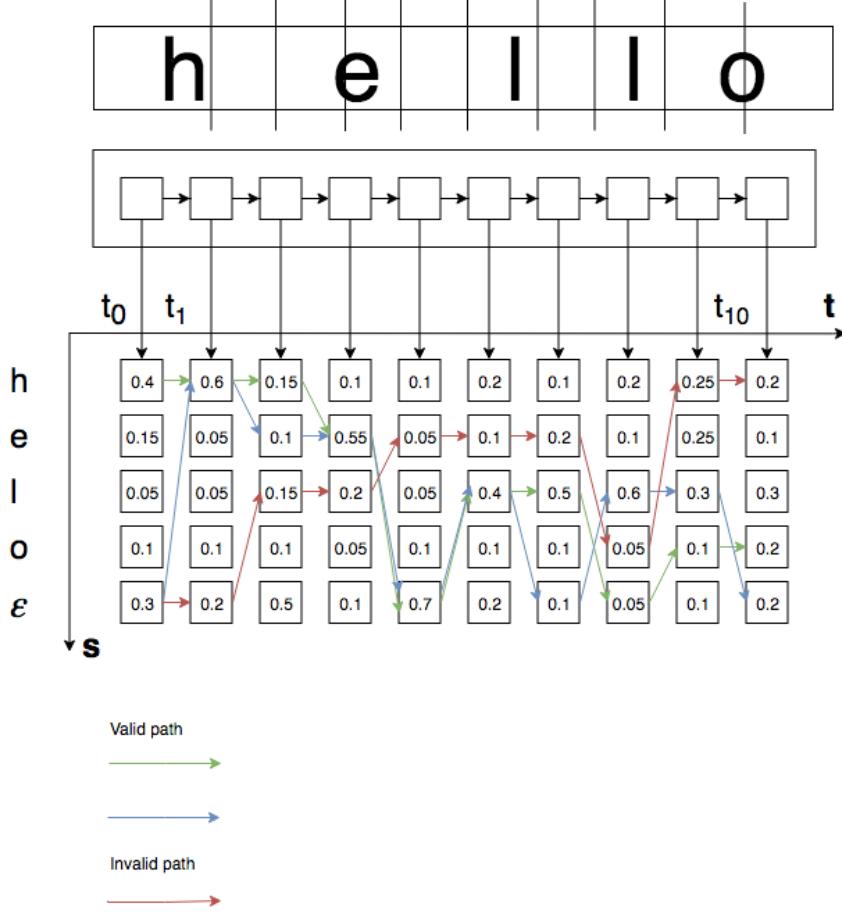
h e | | o

**Hình 3-8:** Quá trình thu giảm chuỗi dự đoán

Mục tiêu của hàm lỗi CTC là tìm ra được các chuỗi mà có xác suất cao nhất trong số các chuỗi hợp lệ. Điều này được thể hiện qua phương trình 3.5. Trong đó  $X$  là dữ liệu đầu vào,  $Y$  là chuỗi dự đoán.  $A$  là chuỗi hợp lệ trong tập các chuỗi có thể có  $A_{X,Y}$ .  $\prod_{t=1}^T p_t(a_t|X)$  là tích xác suất của từng ký tự ứng với vị trí xuất hiện  $t$ .

$$p(Y|X) = \sum_{A \in AX, Y} \prod_{t=1}^T p_t(a_t|X) \quad (3.5)$$

Các bước để tính giá trị của hàm lỗi CTC như theo định nghĩa sẽ qua các bước như hình 3-9. Đầu tiên dữ liệu ảnh đầu vào sẽ qua các lớp CNN để trích xuất đặc trưng. Các đặc trưng này sau đó được qua lớp mạng RNN để đưa ra giá trị dự đoán ứng với từng cột trên đặc trưng. Sau đó các giá trị dự đoán này sẽ được chuyển về chuỗi xác suất nhờ vào hàm Softmax. Lưu ý chiều rộng của đặc trưng không nhất thiết phải bằng số lượng ký tự. Do đó sẽ có một số cột liền nhau đại diện cho một ký tự nào đó hoặc có thể là ký tự trống. Như ta thấy trong hình 3-9 ảnh đầu vào gồm 10 cột và mỗi cột đại diện cho một vài ký tự trong nhãn.



**Hình 3-9: Minh họa cách tính giá trị hàm lỗi CTC**

### 3.1.3 Quá trình tối ưu hóa

Để tìm được cực tiểu cho hàm lỗi, trong quá trình huấn luyện, dữ liệu sẽ lần lượt được đưa vào mạng học và các hệ số của mạng sẽ được hiệu chỉnh lại. Quá trình tối ưu thường được sử dụng nhất là Gradient Descent. Theo đó, mỗi mỗi bước lặp qua tập dữ liệu  $t$ , trọng số sẽ được cập nhật thêm một lượng  $\Delta W$ , nghĩa là  $W^{t+1} = W^t + \Delta W^t$ . Lượng cập nhật này tương ứng với hàm lỗi được thể hiện như sau:

$$\Delta E = \Delta W^T \nabla E(W)$$

Từ công thức trên có thể thấy khi vector gradient  $\nabla E$  gần về 0, hàm lỗi cũng dần về 0. Khi ấy hàm lỗi đã đạt được giá trị cực tiểu địa phương. Giải thuật sẽ dừng khi đạt đến một số lượng lặp nhất định, hoặc các điều kiện dừng đặc biệt khác.

## 3.2 Mạng nơ-ron tích chập

Trong những năm gần đây, mô hình mạng tích chập (Convolutional Neural Network-CNN) nhận được rất nhiều sự quan tâm của cộng đồng nghiên cứu thị giác máy tính. Mạng CNN được đề xuất lần đầu bởi LeCun [19]. Ưu điểm lớn nhất của mạng CNN là có khả năng hoạt động cực kỳ tốt trên các bài toán thị giác máy tính và có thể dễ dàng được mở rộng cho các bài toán khác.

### 3.2.1 Tính chất mạng nơ-ron tích chập

Mạng nơ-ron tích chập có cách tính toán tương tự như các mặt nạ tích chập thông thường. Phép toán tích chập được định nghĩa bằng công thức sau:

$$(f * g)(t) = \int_0^t f(\tau) * g(t - \tau) d\tau$$

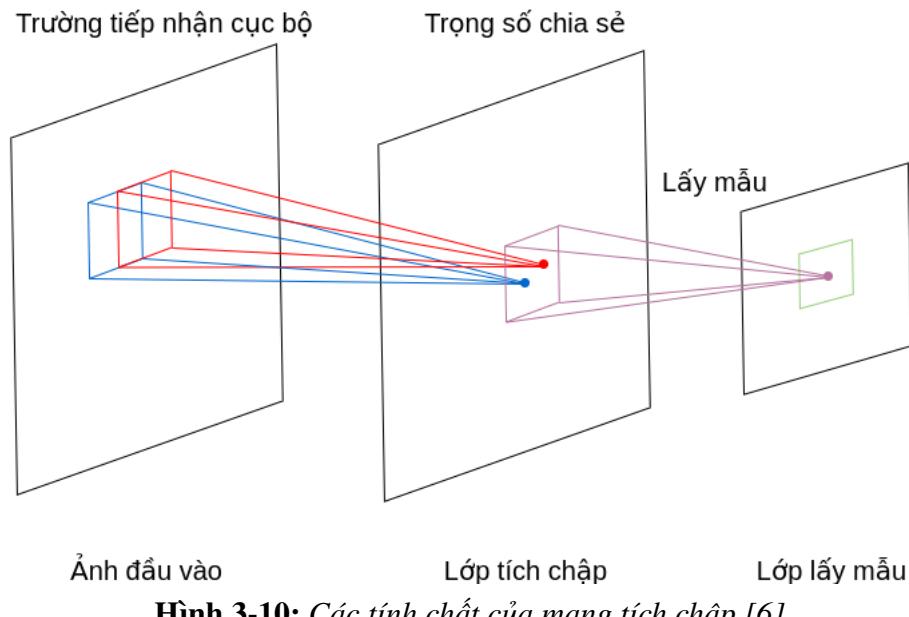
Trong miền rời rạc công thức 3.2.1 được viết lại như sau:

$$(f * g)[n] = \sum_{i=-I}^I f[n-i] * g[i]$$

Với ảnh số công thức 3.2.1 được mở rộng như sau:

$$(f * g)[m, n] = \sum_{j=-J}^J \sum_{i=-I}^I f[m-i, n-j] * g[i, j]$$

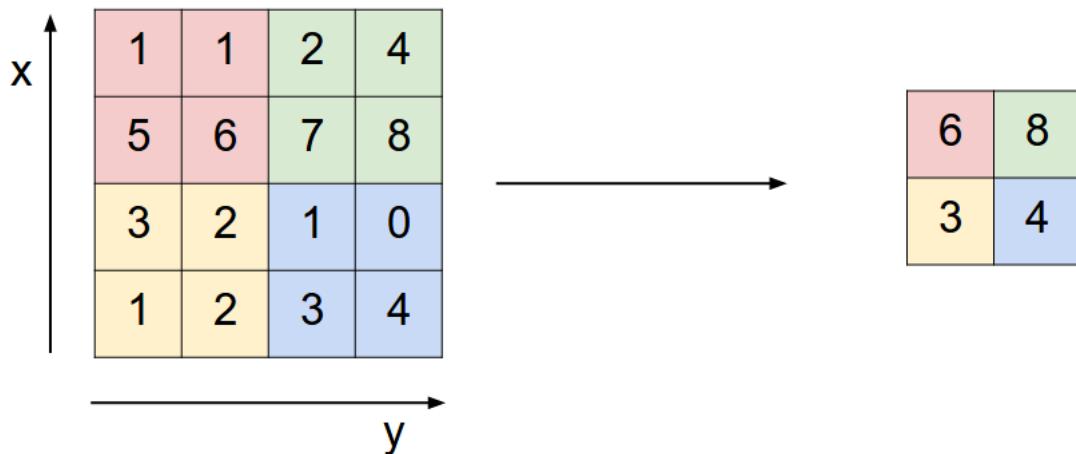
Đây chính là điểm làm nên sự khác biệt chính giữa một mạng tích chập và một mạng nơ-ron kết nối đầy đủ kinh điển. Trong một mạng kết nối đầy đủ thông thường, mỗi nút đầu vào đều có một trọng số riêng khi tính nút ẩn. Tuy nhiên, đối với mạng tích chập thì các trọng số đó là sử dụng chung (shared). Nhờ đó mà việc huấn luyện mạng tích chập trở nên hiệu quả hơn. Với mạng tích chập nhiều lớp, mỗi điểm ứng trên bản đồ đặc trưng của lớp sau sẽ tương ứng với một vùng trên bản đồ đặc trưng trước đó. Nhờ đó mạng tích chập nhiều lớp có thể trích xuất được các đặc trưng từ cơ bản đến trừu tượng. Hơn nữa các cửa sổ dùng để rút trích đặc trưng là được học một cách tự động. Trong khi các cách trích xuất đặc trưng thủ công với số lượng cửa sổ ít và cố định thì rõ ràng mạng CNN nhiều lớp có ưu thế hơn hẳn.



**Hình 3-10:** Các tính chất của mạng tích chập [6]

### 3.2.2 Lấy mẫu (Subsampling hay pooling)

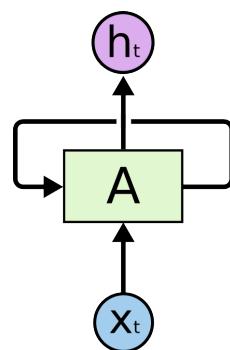
Mạng CNN thường không bắt biền với phép xoay và phép co giãn. Do đó để mạng học sâu có thể làm việc tốt và tổng quát cao, các lớp lấy mẫu thường được thêm vào xen kẽ giữa các lớp CNN. Với đầu vào là bản đồ đặc trưng có được từ lớp trước, ta tiến hành lấy mẫu theo từng cửa sổ  $2 \times 2$  hoặc  $3 \times 3$ , v.v. Việc lấy mẫu được tiến hành theo cách lấy giá trị lớn nhất hoặc trung bình của cửa sổ. Như hình 3-11 mô tả lấy mẫu theo hàm max trên một ma trận  $4 \times 4$ . Cửa sổ được sử dụng là cửa sổ  $2 \times 2$  với bước trượt là 2. Với mỗi lần trượt cửa sổ, hàm lấy mẫu sẽ lấy giá trị lớn nhất trong vùng  $2 \times 2$ . Kết quả sau khi lấy mẫu là ma trận có kích thước  $2 \times 2$  như hình 3-11. Lấy mẫu có tác dụng làm giảm kích thước đầu ra và giữ lại những giá trị quan trọng.



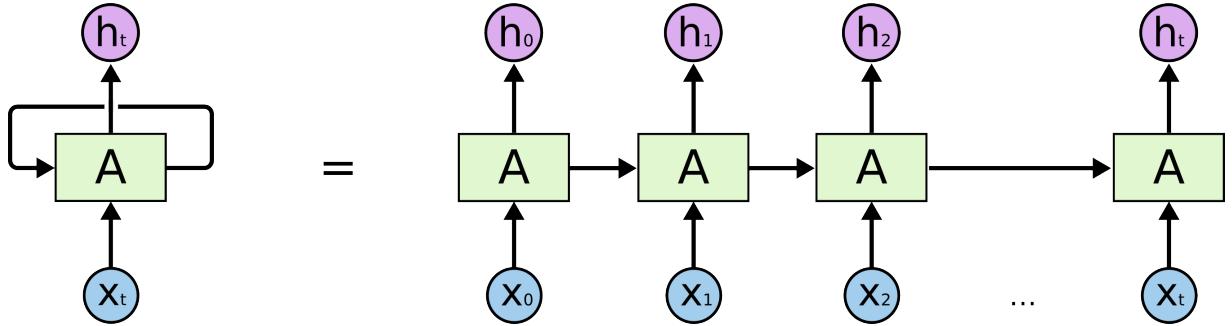
**Hình 3-11:** Minh họa về lớp Max Pooling với kích thước của sô<sup>2</sup> 2x2 và độ dịch chuyển là 2

### 3.3 Mạng nơ-ron đệ quy

Mạng nơ-ron đệ quy (Recurrent Neural Network-RNN) là một loại mạng nơ-ron trong đó mỗi nốt trong các lớp ẩn có một kết nối với chính bản thân. Chính kết nối này tạo ra các trạng thái nội tại của kiến trúc mạng cho phép mô hình hoá các chuỗi với độ dài bất kỳ. Mạng hồi quy thường được biểu diễn dưới dạng lặp như hình 3-12, tuy nhiên để dễ hình dung mạng nơ-ron hồi quy cũng có thể được biểu diễn dưới dạng không lặp như hình 3-13.



**Hình 3-12:** Mạng nơ-ron đệ quy biểu diễn dưới dạng lặp



**Hình 3-13:** Mạng nơ-ron đê quy biểu diễn dưới dạng không lặp

Một mạng nơ-ron đê quy có thể nhận vào một chuỗi có chiều dài bất kỳ và tạo ra một chuỗi nhãn có chiều dài tương ứng nên kiến trúc này rất phù hợp với bài toán nhận diện chuỗi, theo vết đối tượng. Như mô tả trong hình 3-13, tại mỗi thời điểm  $t$  một RNN điển hình  $A$  sẽ tạo ra một vector  $h_t$  chứa toàn bộ thông tin của các dữ liệu đầu vào từ  $X_0$  tới  $X_t$ . Công thức tổng quát của mạng RNN được định nghĩa như sau:

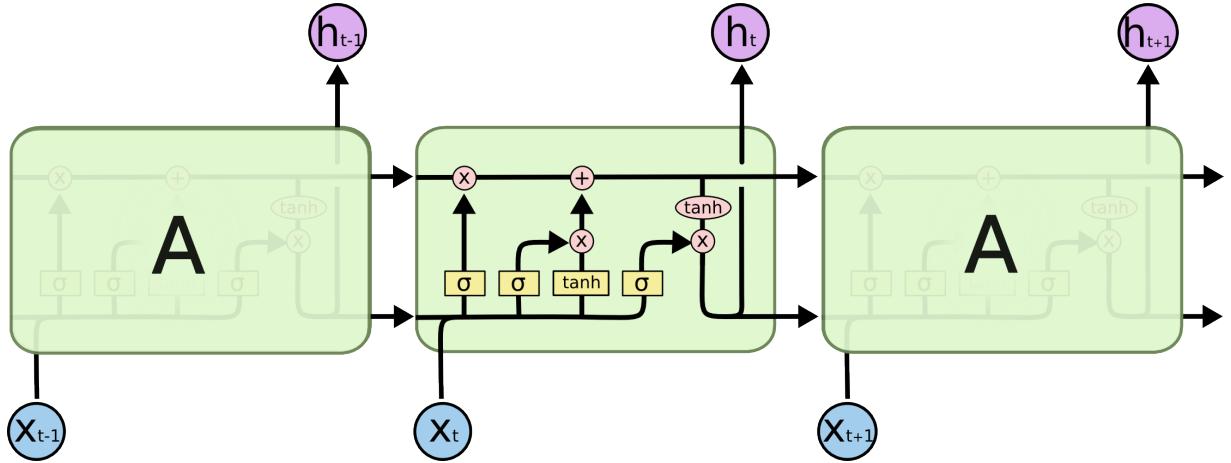
$$h_t = f_W(h_{t-1}, x_t) \quad (3.6)$$

Với  $h_t$  là trạng thái ẩn,  $f_W$  là một hàm số với tham số  $W$ ,  $x_t$  là dữ liệu đầu vào của mạng. Trong trường hợp hàm  $f_W$  là hàm  $\tanh$  ta có công thức như sau:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)y_t = Why_t \quad (3.7)$$

Tuy nhiên, kiến trúc RNN cổ điển rất khó để áp dụng trong thực tế vì vấn đề liên quan đến việc mất mát hoặc bùng nổ giá trị (vanishing and exploding gradient) được dùng để cập nhật các trọng số của mạng thông qua quá trình học khi phải mô hình hóa các chuỗi rất dài.

Long Short Term Memory (LSTM) [15], một biến thể nổi tiếng của mạng nơ-ron hồi quy, được đề xuất như là một giải pháp cho vấn đề vừa được nêu trên. Điểm chính trong kiến trúc mạng của LSTM chính là các ô nhớ với các cổng cho phép lưu trữ hoặc truy xuất thông tin. Các cổng này cho phép ghi đè (input gate), loại bỏ dư thừa (forget gate) và truy xuất (output gate) các thông tin được lưu trữ bên trong các ô nhớ. Cấu trúc của một phần tử LSTM và công thức tổng quát được mô tả như sau:



**Hình 3-14:** Cấu trúc của mạng LSTM

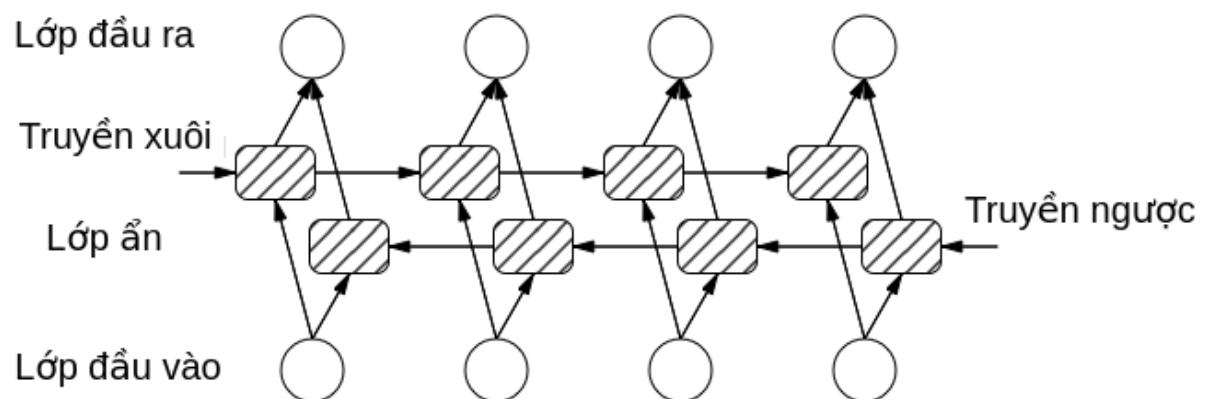
Về mặt toán học, một phần tử LSTM được định nghĩa như sau:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

Với một số bài toán nhận dạng chuỗi thông thường để dự đoán cho ký tự hiện tại thường dùng đến thông tin của các ký tự trước đó. Tuy nhiên để đảm bảo độ chính xác, thông tin của các ký tự phía sau cũng được sử dụng đến. Để làm được điều đó, mô hình Bidirectional LSTM (BiLSTM) được đề xuất. Đây là một mạng LSTM gồm hai phần tử hoạt động độc lập. Một phần tử sẽ mô hình hóa chuỗi dữ liệu theo hướng từ trái sang phải và phần tử còn lại sẽ làm theo chiều ngược lại.



**Hình 3-15:** Mạng Bidirectional LSTM

# CHƯƠNG 4: MÔ HÌNH ĐỀ XUẤT

## 4.1 Tổng quan

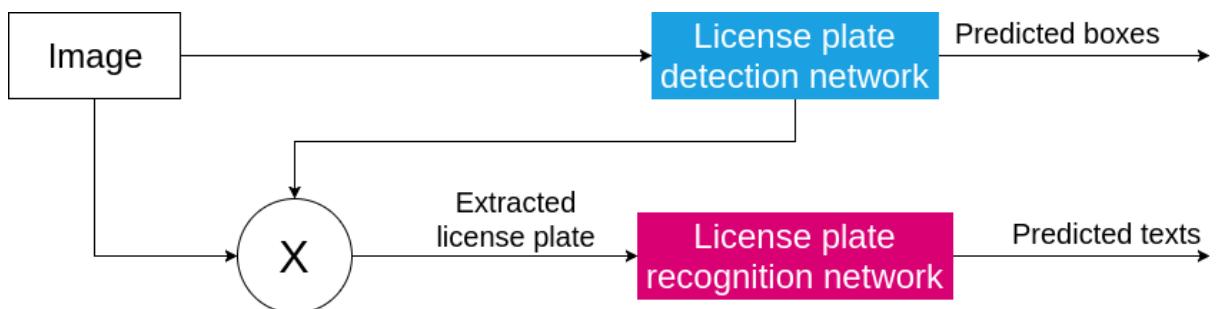
Dựa trên ý tưởng của công trình [5] và [21], đề tài xây dựng một mô hình tích hợp giữa phát hiện và nhận dạng biển số xe. Mô hình này bỏ qua bước phân đoạn ký tự và nhận diện trực tiếp chuỗi ký tự từ ảnh biển số hoặc từ bản đồ đặc trưng của phần mạng phát hiện. Mô hình gồm hai mạng chính: mạng phát hiện và mạng nhận dạng. Trong đó mạng phát hiện sử dụng kiến trúc mạng SSD [3] với kích thước đầu vào là 640x640, mạng nền là ResNet-50 hoặc ResNet-34. Mạng nhận diện sử dụng kiến trúc CRNN [1]. Tùy theo các thí nghiệm khác nhau mà đề tài sẽ có những thay đổi kiến trúc của từng mạng phát hiện và nhận dạng. Kiến trúc mô hình được đề xuất trong đề tài là kiến trúc tuần tự.

Kiến trúc tuần tự được mô tả theo như hình 4-1. Trình tự các bước từ ảnh đầu vào cho tới kết quả đầu ra như sau:

- **Bước 1:** Đầu tiên ảnh đầu vào sẽ được cho qua lớp mạng CNN để trích xuất đặc trưng. Ảnh đầu vào là ảnh màu (BGR) và có kích thước bất kỳ. Trước khi đưa vào lớp trích xuất đặc trưng ảnh sẽ được đưa về kích thước 640x640. Tại đầu ra của lớp này, ảnh sẽ biến đổi thành các bản đồ đặc trưng với kích thước lần lượt là (160x160x64), (80x80x128), (40x40x256), (20x20x512), (10x10x128), (5x5x256).
- **Bước 2:** Các bản đồ đặc trưng từ bước một gồm 6 kích thước khác nhau sẽ được đưa qua lần lượt các lớp mạng Multibox của SSD [3] để phát hiện vị trí của biển số xe cũng như phân loại biển số. Chi tiết về bước phát hiện biển số xe cũng như kiến trúc mạng sẽ được trình bày trong mục 4.2
- **Bước 3:** Tại bước này vùng biển số xe sẽ được cắt ra từ ảnh đầu vào. Sau đó các bản đồ đặc trưng cho vùng biển số này, tùy vào loại biển số là ngắn hay dài sẽ được xử lý tương ứng và đưa về kích thước chung là 32x100. Vùng biển số sau xử lý sẽ được đưa vào mạng nhận diện phía sau. Mục 4.3 sẽ trình bày chi tiết về cách trích xuất

biển số xe.

- **Bước 4:** Đây là bước quan trọng nhất trong toàn bộ hệ thống. Vùng biển số sau khi được trích xuất sẽ đưa qua lớp mạng nhận diện để ra chuỗi ký tự trực tiếp mà không cần qua bước phân đoạn ký tự. Mạng nhận diện ở mô hình tích hợp là mạng CRNN [1]. Chi tiết về bước phát hiện được trình bày trong mục 4.4.
- **Bước 5:** Bước cuối cùng trong mô hình là bước tổng hợp các kết quả từ hai lớp mạng phát hiện và nhận diện để cho kết quả sau cùng. Tại bước này các vùng biển số sẽ được lọc bớt dựa theo kết quả của việc nhận diện.



**Hình 4-1:** Kiến trúc tổng quan của mô hình tuân tự đề xuất

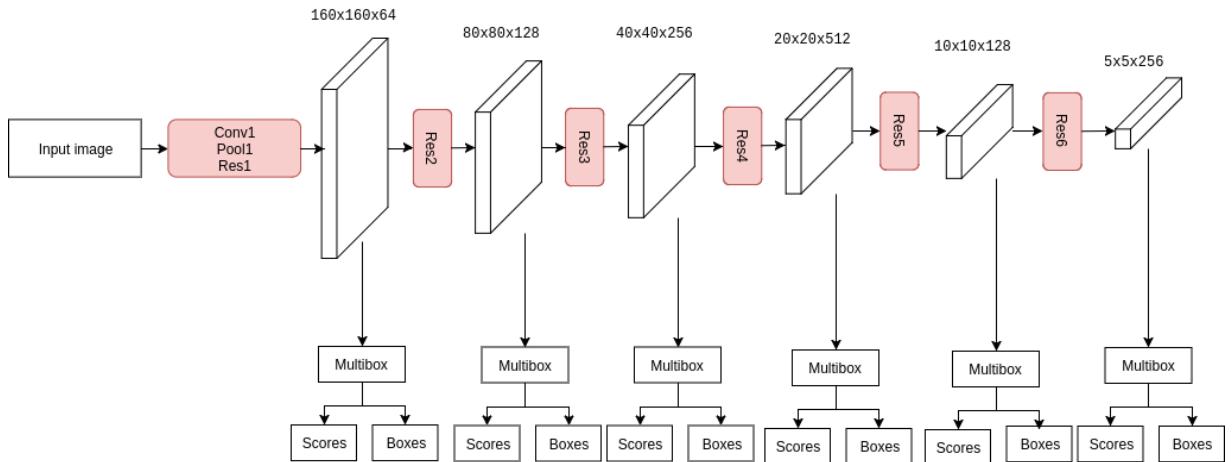
## 4.2 Phát hiện biển số xe

Để phát hiện biển số xe đề tài dựa trên kiến trúc SSD [3]. Tuy nhiên, đề tài có một số thay đổi chính như mạng nền, kích thước đầu vào của ảnh, số lớp Multibox và tỷ lệ các ô chuẩn. Trong đó mạng nền được thay đổi từ VGG [14] sang ResNet [20]. Hai cấu hình ResNet chính được sử dụng là ResNet-34 và ResNet-50. Kích thước ảnh đầu vào trong mạng SSD gốc là 300x300. Tuy nhiên với bài toán phát hiện biển số xe, ảnh đầu vào thường có độ phân giải lớn. Do đó đề tài chọn kích thước ảnh đầu vào là 640x640 để đảm bảo biển số không bị thu nhỏ quá nhiều dẫn tới việc phát hiện kém đi.

Để đảm bảo các ô chuẩn (default box) trùng khớp tốt nhất với cả hai dạng biển số xe, đề tài đã thiết kế lại các ô chuẩn bằng cách thay đổi tỷ lệ giữa chiều rộng và chiều cao. Đề tài đề xuất sử dụng 3 tỷ lệ kích thước ô chuẩn như sau: 1:1, 1.55:0.8 và 1.02:0.99.

Ngoài trừ ô chuẩn có kích thước 1:1, để xác định các tỷ lệ còn lại, đề tài dựa trên thống kê về tỷ lệ trung bình của mỗi loại biển số xe.

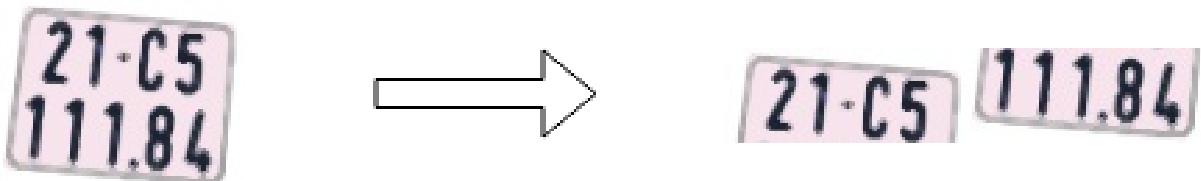
Sau cùng kiến trúc mạng phát hiện đề tài đề xuất như hình: 4-2:



**Hình 4-2: Kiến trúc của mạng phát hiện biển số xe**

### 4.3 Trích xuất vùng biển số xe

Với kiến trúc tuần tự vùng biển số sẽ được cắt ra từ ảnh gốc. Vùng biển số được cắt ra và dựa trên loại biển số là ngắn hay dài mà sẽ được chia làm đôi. Với biển số ngắn sau khi được chia làm đôi sẽ được ghép nối tiếp với nhau theo chiều ngang. Sau đó tất cả biển số sẽ được đưa về kích thước cố định là 32x100. Hình 4-3 minh họa cho thao tác trích xuất vùng biển số xe.



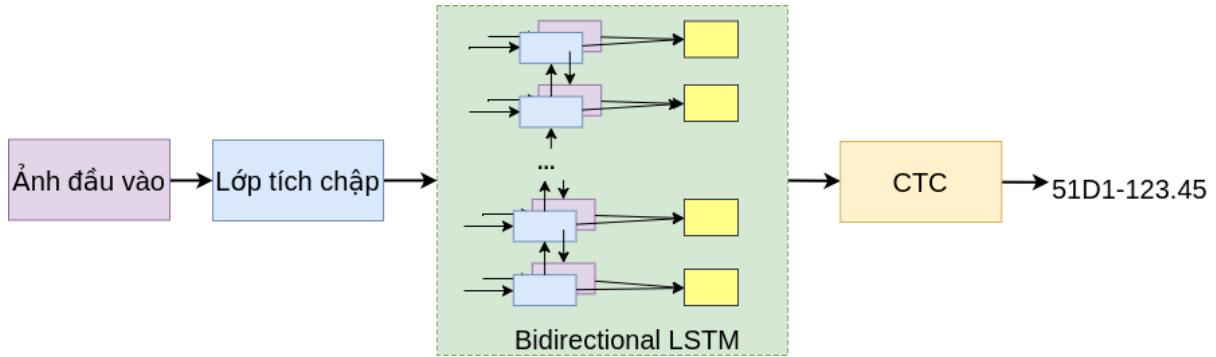
**Hình 4-3: Biển số sau khi cắt ra sẽ được ghép lại và đưa về kích thước chung**

Hai thao tác chính là cắt vùng ảnh và co giãn kích thước được sử dụng như một lớp mạng để thuận tiện hơn cho thao tác huấn luyện và kiểm thử sử dụng GPU. Hai lớp mạng

thích hợp nhất cho thao tác này là Adaptive Max Pooling và Roi Pooling [9].

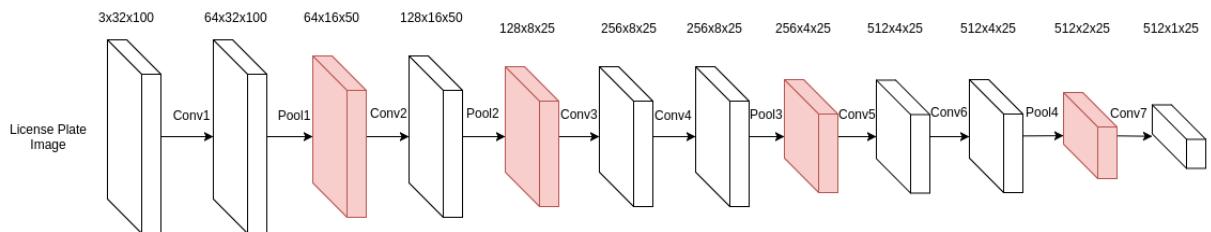
#### 4.4 Nhận dạng biển số xe

Để nhận dạng chuỗi biển số, đề tài dựa trên kiến trúc mạng mà công trình [1].



**Hình 4-4:** Kiến trúc tổng quan của mạng CRNN sử dụng trong mô hình đề xuất

Trong đó các lớp mạng tích chập dùng để rút trích đặc trưng là kiến trúc mạng tích chập đầy đủ trong đó bao gồm các lớp tích chập đan xen là lớp lấy mẫu. Hình 4-5 minh họa cho cấu trúc này:



**Hình 4-5:** Minh họa cho các lớp tích chập được sử dụng trong mạng CRNN

Chi tiết về cấu hình các lớp tích chập và lấy mẫu của mạng CRNN được sử dụng trong mô hình đề xuất:

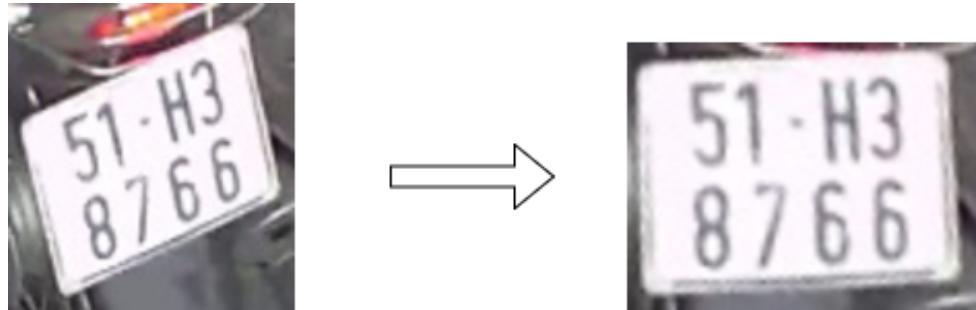
**Bảng 4-1: Cấu hình các lớp tích chập trong mạng CRNN[1]**

Khối	Số kênh đầu vào	Số kênh đầu ra	Kích thước nhân	Chèn thêm (Padding)	Bước (Stride)	Chuẩn hóa bó (Batchnorm)
Conv_1	3	100	(3, 3)	(1, 1)	(1, 1)	Không
Pool_1	100	50	(2, 2)	(0, 0)	(2, 2)	-
Conv_2	50	50	(3, 3)	(1, 1)	(1, 1)	Không
Pool_2	50	25	(2, 2)	(0, 0)	(2, 2)	-
Conv_3	25	25	(3, 3)	(1, 1)	(1, 1)	Có
Conv_4	25	25	(3, 3)	(1, 1)	(1, 1)	Không
Pool_3	25	25	(2, 2)	(0, 1)	(2, 1)	-
Conv_5	25	25	(3, 3)	(1, 1)	(1, 1)	Có
Conv_6	25	25	(3, 3)	(1, 1)	(1, 1)	Không
Pool_4	25	25	(2, 2)	(0, 1)	(2, 1)	-
Conv_7	25	25	(2, 2)	(0, 0)	(1, 1)	Không

Các lớp tích chập đều sử dụng hàm kích hoạt là hàm ReLU hoặc Leaky ReLU. Chú ý bên cạnh các lớp tích chập và lấy mẫu xen kẽ nhau để giảm kích thước bản đồ đặc trưng, các lớp lấy mẫu như Pool\_3 và Pool\_4 có bước trượt cũng như phần chèn thêm không đều nhau theo hai chiều. Đặc biệt lớp cuối cùng lớp Conv\_7 có bước trượt bằng 1 và không chèn thêm. Những thay đổi trên cho phép bản đồ đặc trưng sau cùng có chiều cao chỉ còn bằng 1, cụ thể kích thước bản đồ đặc trưng sẽ là 512x1x25. Ngoài ra việc thêm bước chuẩn hóa bó (batchnorm) vào các lớp Conv\_3 và Conv\_5 cũng giúp mạng ổn định hơn, tránh hiện tượng giá trị đạo hàm rơi về 0 hoặc  $\pm\infty$ .

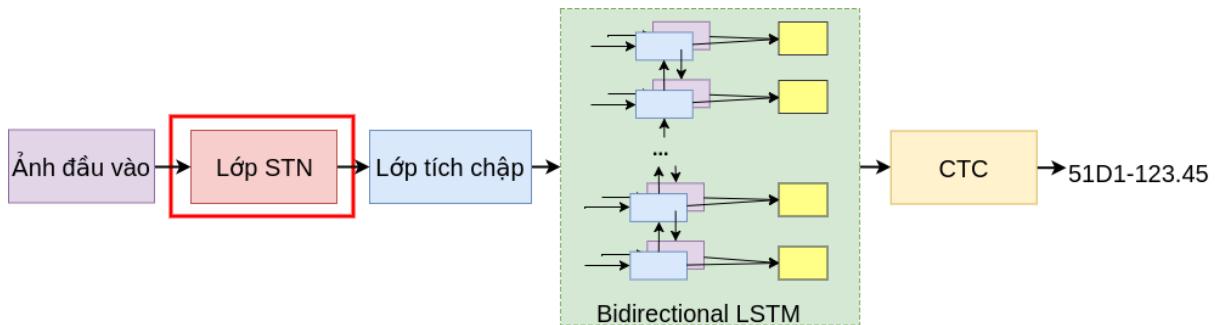
Nhận thấy với dữ liệu biển số được biến đổi phôi cảnh để về dạng thẳng góc như hình 4-6 giúp cải thiện độ chính xác của việc nhận diện. Đề tài đã tìm cách cải thiện mô hình CRNN ban đầu theo hướng giúp mạng học được phép biến đổi không gian cho ảnh đầu

vào trước khi đi cho qua tiếp các lớp mạng nhận diện sau.



**Hình 4-6:** Dữ liệu đầu vào sau khi qua phép biến đổi phối cảnh

Để làm được điều đó, đề tài sử dụng lớp mạng biến đổi không gian (Spatial Transformation Network - STN) từ công trình [22] thêm vào trước các lớp tích chập của mạng CRNN gốc như hình 4-7



**Hình 4-7:** Kiến trúc cải tiến của mạng CRNN

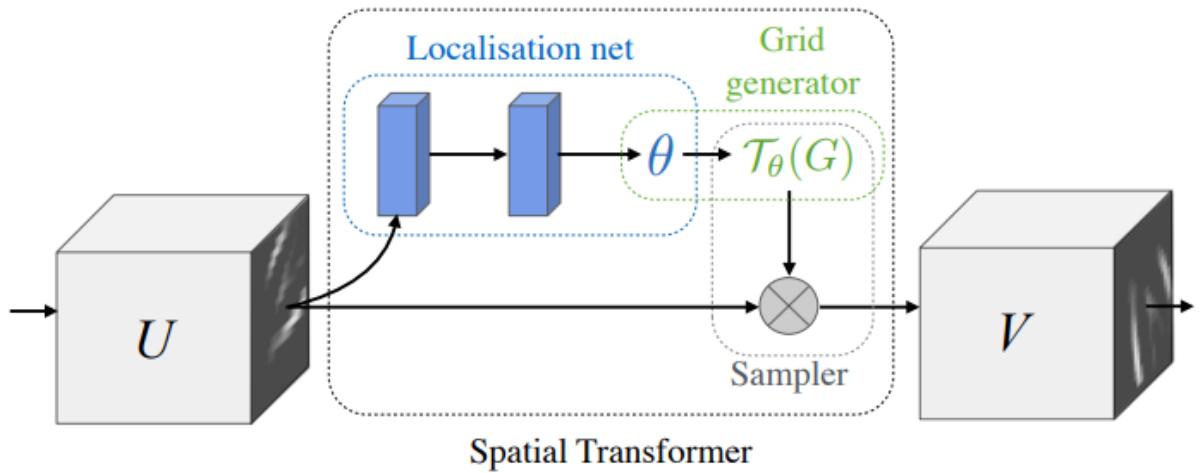
Mạng STN gồm 3 phần chính:

- Localization network: nhận vào bản đồ đặc trưng và dự đoán ma trận biến đổi affine.

$$A_\theta = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix}$$

- Grid generator: tạo ra lưới với kích thước định sẵn từ ma trận biến đổi affine.
- Bilinear sampler: tạo ra bản đồ đặc trưng sau cùng từ lưới đã tạo ra dùng phép nội suy song tuyến (bilinear interpolation).

Hình 4-8 mô tả cấu trúc của lớp mạng STN:



**Hình 4-8:** Minh họa cho các lớp tích chập được sử dụng trong mạng CRNN

# CHƯƠNG 5: THÍ NGHIỆM

Từ mô hình đã nêu ở phần trên, để tài thực hiện một vài thí nghiệm nhằm đánh giá độ hiệu quả của mô hình cũng như tìm ra được các siêu tham số phù hợp nhất với mô hình đề ra. Trình tự các bước được mô tả như sau.

## 5.1 Tiêu chí đánh giá

Phương pháp đề xuất sẽ tập trung vào đánh giá việc phát hiện biến số dựa trên độ đo chính là Average Precision (AP).

Độ đo AP được định nghĩa là độ chính xác trung bình của một tập 11 giá trị truy hồi  $r \in \{0, 0.1, 0.2, \dots, 1.0\}$ :

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, 0.2, \dots, 1.0\}} \hat{p}(r) \quad (5.1)$$

Trong đó độ chính xác tại giá trị truy hồi thứ  $i$  được lấy là độ chính xác tối đa tại những giá trị truy hồi  $r'$  lớn hơn  $r$ .

$$\hat{p}(r) = \max_{r' \geq r} p(r') \quad (5.2)$$

mAP là trung bình AP của từng lớp.

Độ trùng khớp giữa 2 vùng bao chữ nhật:

$$m_a(r, r') = \frac{a(r \cap r')}{a(r \cup r')} \quad (5.3)$$

Trong đó  $a(r)$  là diện tích vùng chữ nhật  $r$ . Giá trị trùng khớp tốt nhất cho một vùng chữ nhật  $r$  trong tập chữ nhật  $R$  được định nghĩa:

$$m(r, R) = \max(m_a(r, r')) \forall r \in R \quad (5.4)$$

Độ chính xác và độ truy hồi được định nghĩa:

$$p = \frac{\sum_{r_e \in E} m(r_e, T)}{|E|} \quad (5.5)$$

$$r = \frac{\sum_{r_t \in T} m(r_t, E)}{|T|} \quad (5.6)$$

Trong đó E và T tương ứng là tập hợp các đường bao chữ nhật ước lượng và ground truth.  $r_t$  và  $r_e$  là các đường bao chữ nhật thuộc các tập hợp trên.

Độ chính xác nhận diện một chuỗi:

$$WRA = \frac{|C|}{|T|} \quad (5.7)$$

Trong đó C và T là số lượng từ nhận diện đúng và số lượng từ ground truth. Với các phương pháp tích hợp độ trùng khớp  $m_a$  phải lớn hơn hoặc bằng 0.5 và chuỗi nhận diện được phải khớp hoàn toàn [23].

## 5.2 Chuẩn bị dữ liệu

Các tập dữ liệu biển số xe không quá phổ biến. Một vài tập dữ liệu tiêu biểu có thể kể đến như: CCPD [2], UFPR-ALPR [24], AOLP [25] và PKUData [26]. Tuy nhiên các tập dữ liệu này là của nước ngoài có kích thước, bố trí khác so với Việt Nam. Các tập dữ liệu này có thể được dùng cho việc huấn luyện phát hiện biển số xe. Dữ liệu chính để sử dụng cho đề tài chủ yếu là từ việc thu thập ngoài thực tế và sinh tự động. Việc thu thập dữ liệu thực tế sẽ được tiến hành trong những điều kiện khác nhau như: góc độ của camera, điều kiện chiếu sáng, độ phân giải của camera, .... Tập dữ liệu sẽ được chia làm 2 tập riêng biệt là tập huấn luyện và tập kiểm thử. Tập huấn luyện với mục đích tối ưu các thông số của hệ thống nhận diện. Tập kiểm thử dùng để đánh giá kết quả của phương pháp sau quá trình huấn luyện. Cả 2 tập dữ liệu đều sẽ được gán nhãn vùng có biển số xe xuất hiện và các ký tự trong các biển số đó.

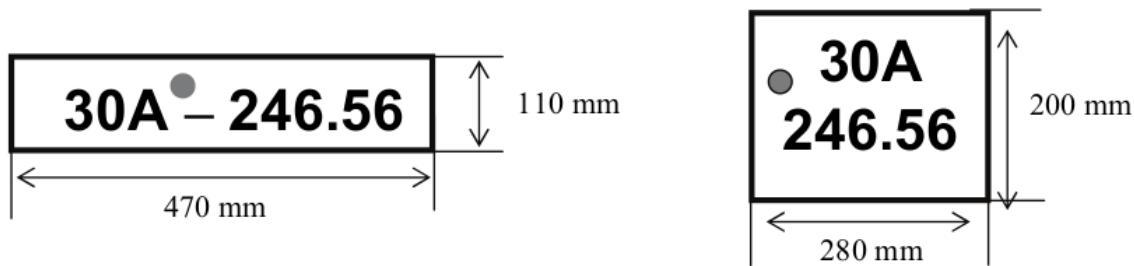
### 5.2.1 Quy định về biển số xe Việt Nam

Dựa trên những quy định của Thông tư 15/2014/TT-BCA, tùy thuộc vào loại xe, tổ chức sở hữu, vùng lãnh thổ mà sẽ có những quy định riêng về kích thước, cách bố trí cũng như nội dung của biển số khác nhau. Tuy nhiên để tài chỉ tập trung vào các loại

biển số thông thường của ô tô và xe máy. Nếu xét về kích thước, biển số có thể được chia thành 2 loại là biển số ngắn và biển số dài. Xét về tổng số lượng ký tự có trong biển số, biển số thông thường có 8 hoặc 9 ký tự. Một số quy định về kích thước và bố cục của biển số xe:

### **Biển số xe ô tô:**

- Kích thước biển số xe ô tô gồm 470x110 và 280x200 được mô tả như hình sau:

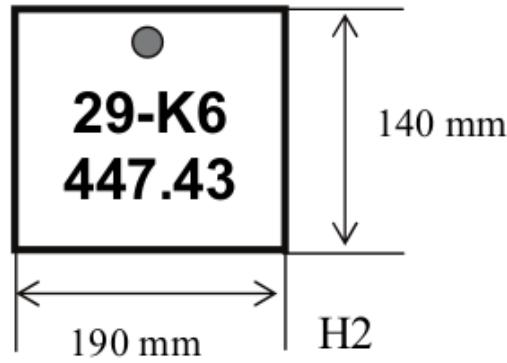


**Hình 5-1: Các kích thước biển số xe ô tô**

- Biển số gồm các nhóm chữ và số sắp xếp cụ thể như sau: Hai số đầu là ký hiệu địa phương và sê ri đăng ký xe. Nhóm số thứ hai là thứ tự xe đăng ký gồm 05 chữ số tự nhiên, từ 000.01 đến 999.99
- Về kích thước của chữ và số:
  - Chiều cao của chữ và số: 80 mm
  - Chiều rộng của chữ và số: 40 mm
  - Nét đậm của chữ và số: 10 mm.
  - Khoảng cách giữa các chữ và số: 10 mm. Riêng khoảng cách giữa các số với số 1 là 20 mm.

### **Biển số xe máy:**

- Kích thước biển số xe máy gồm 190x140 được mô tả như hình sau:



**Hình 5-2:** Kích thước biển số xe ô tô

- Biển số gồm các nhóm chữ và số sắp xếp cụ thể như sau: Hai số đầu là ký hiệu địa phương và sê ri đăng ký xe. Giữa ký hiệu địa phương và sê ri đăng ký được phân cách bằng dấu gạch ngang (-). Nhóm số thứ hai là thứ tự xe đăng ký gồm 05 chữ số tự nhiên, từ 000.01 đến 999.99.
- Về kích thước của chữ và số:
  - Chiều cao của chữ và số: 55 mm
  - Chiều rộng của chữ và số: 22 mm
  - Nét đậm của chữ và số: 7 mm.
  - Khoảng cách giữa các chữ và số: 5 mm. Khoảng cách giữa các số thứ tự đăng ký là 12 mm.

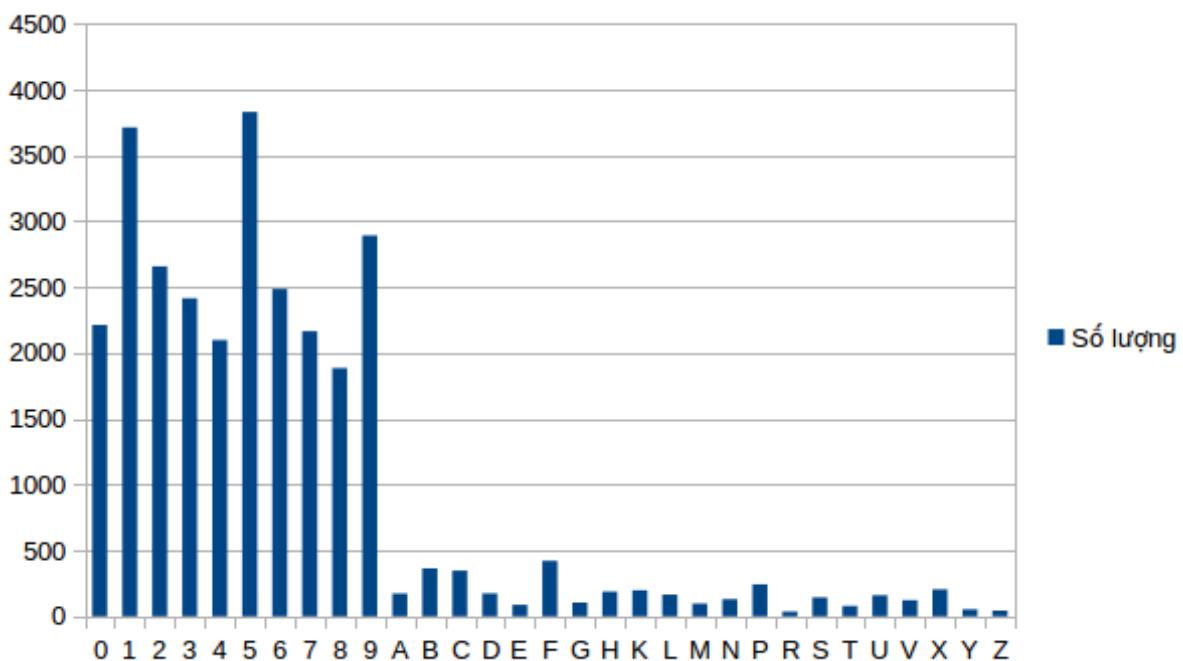
### 5.2.2 Xây dựng tập biển số từ dữ liệu thực tế

Tập dữ liệu thực tế được thu thập từ các nguồn chính là từ bãi giữ xe trường Đại học Bách Khoa TP.HCM, các video tự thu thập trên đường phố và trên Internet. Với tập bãi giữ xe, dữ liệu biển số là của xe máy. Góc quay camera được đặt cố định và mỗi ảnh chỉ có một biển số xe. Với tập dữ liệu tự quay, camera không có góc quay cố định và mỗi ảnh xuất hiện nhiều biển số xe.

Một số thông tin về tập dữ liệu:

- Tổng số lượng ảnh là 3516
- Số lượng biển số xe 5282
- Tổng số lượng các ký tự 29876
- Biển số xe bao gồm các ký tự sau: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, K, L, M, N, P, R, S, T, U, V, X, Y, Z

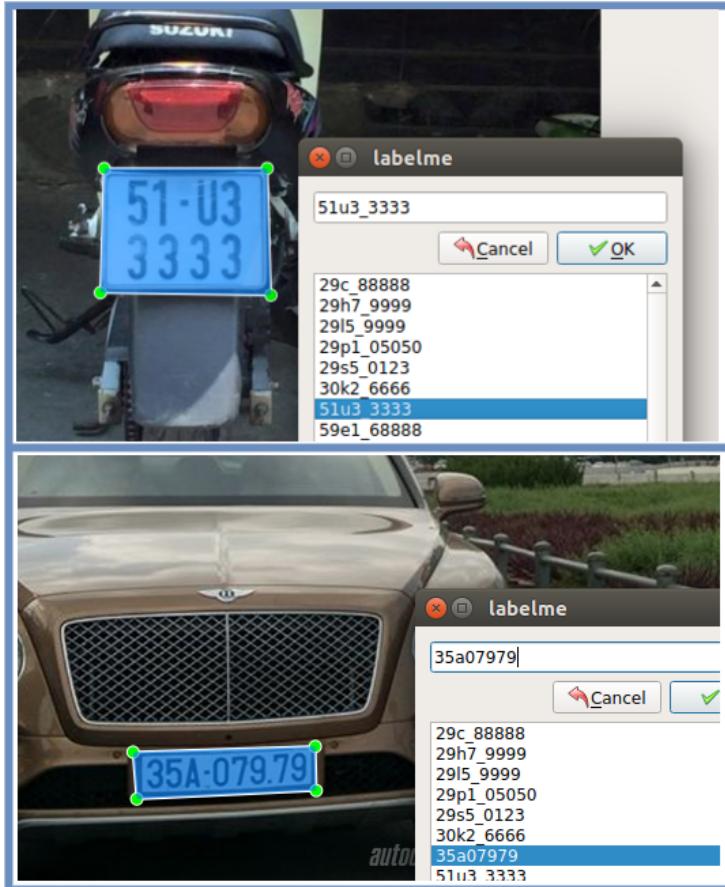
Chi tiết về phân phối của từng ký tự trong tập dữ liệu thật:



**Hình 5-3: Phân phối của từng ký tự trong tập dữ liệu**

Tập dữ liệu thu thập được sẽ được gán nhãn bao gồm thông tin về vị trí và chuỗi ký tự có trong biển số. Vị trí của biển số được xác định bởi tọa độ bốn góc của biển số. Khi huấn luyện mô hình phát hiện, bốn góc này sẽ được chuyển đổi thành đường bao chữ nhật lớn nhất bao trọn vùng biển số. Trong quá trình đánh nhãn vị trí, các đường bao này được đánh sao cho vẫn còn thấy được cạnh của biển số. Điều này nhằm giúp sau quá trình huấn luyện, mô hình sẽ hạn chế nhầm lẫn biển số xe với các đối tượng có chứa chuỗi ký tự khác. Chuỗi ký tự được đánh chỉ bao gồm chữ cái số. Các ký tự khác như dấu gạch

ngang "-" và dấu chấm "." nếu có sẽ được bỏ qua. Để phân biệt giữa biển số ngắn và biển số dài, dấu gạch dưới "\_" sẽ được thêm vào chuỗi ký tự với biển số ngắn. Hình 5-4 minh họa cho quá trình gán nhãn biển số xe.



**Hình 5-4:** Cách gán nhãn cho biển số ngắn và dài

### 5.2.3 Sinh tập biển số

Vấn đề dữ liệu cho quá trình huấn luyện mạng học sâu đóng vai trò quan trọng quyết định độ hiệu quả của mạng. Thông thường dữ liệu huấn luyện khá lớn từ vài trăm ngàn đến hàng triệu ảnh. Để có thể thu thập cũng như gán nhãn bằng tay cho số dữ liệu lớn như vậy là điều khá khăn, tốn nhiều thời gian và công sức. Do đó việc sinh dữ liệu tự động để bổ sung cho dữ liệu huấn luyện là việc cần thiết. Dữ liệu sinh tự động cần phải có các đặc trưng về mặt thống kê tương tự như dữ liệu thực. Quy trình để sinh dữ liệu tự động gồm các bước sau:

- Bước 1: thu thập ảnh nền mẫu từ thực tế cũng như từ các tập dữ liệu công khai trên Internet.
- Bước 2: biển số xe được tạo ra bằng cách tạo phần biển số đơn giản bao gồm nền trắng và viền bao. Chuỗi ký tự được sinh ngẫu nhiên với font chữ có sẵn và theo định dạng đã được quy định ở trên sau đó được chèn vào phần biển số.
- Bước 3: biển số xe sau đó sẽ được điều chỉnh độ sáng tối, độ mờ và thêm nhiễu.
- Bước 4: biển số sẽ được qua phép biến đổi phối cảnh để được gắn vào ảnh nền mẫu.



**Hình 5-5:** Minh họa quá trình sinh biển số ngẫu nhiên

Một vài mẫu biển số sinh tự động như hình 5-6. Trong đó hàng trên là mẫu biển số ngắn và hàng dưới là mẫu biển số dài:



**Hình 5-6:** Một số mẫu biển số xe sinh tự động

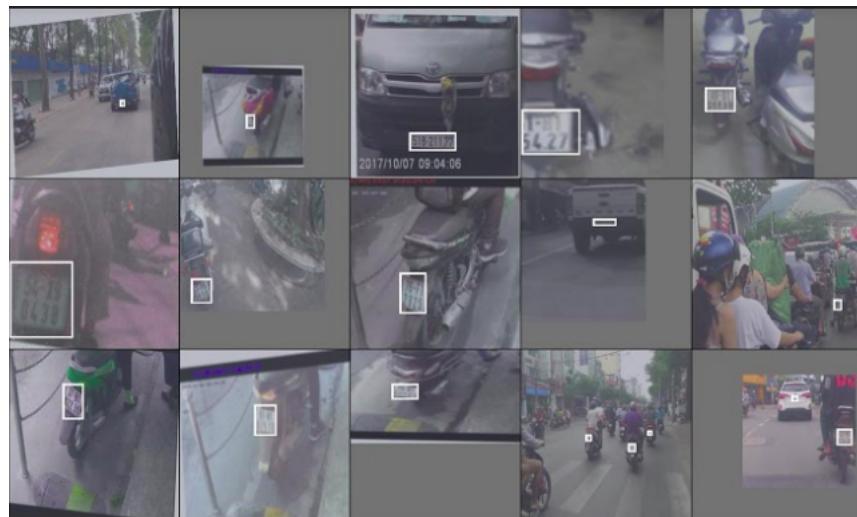
#### 5.2.4 Làm giàu dữ liệu

Trong quá trình huấn luyện dữ liệu sẽ được làm giàu bởi các phép biến đổi màu sắc cũng như các phép biến đổi hình học. Các phép biến đổi chủ yếu được sử dụng như:

- Biến đổi màu ngẫu nhiên

- Biến đổi ảnh xám ngẫu nhiên
- Biến đổi hình học (phép biến đổi xoay và phôi cảnh)
- Cắt ảnh ngẫu nhiên
- Thêm nhiễu và mờ
- Chuẩn hóa ảnh

Trong đó dữ liệu để huấn luyện sẽ được qua đầy đủ các phép biến đổi trên. Với tập đánh giá và kiểm thử ảnh chỉ được chuẩn hóa. Các phép biến đổi đều có xác suất xảy ra lớn hơn 0.5. Tuy nhiên một số phép biến đổi như thêm nhiễu và làm có xác suất thấp hơn trong khoảng 0.1. Với phép biến đổi hình học, góc xoay không quá  $30^0$  và việc biến đổi vẫn giữ nguyên phần biến số không bị mất hoặc bị che khuất. Hình 5-7 minh họa một số ảnh đầu vào đã qua các phép biến đổi làm giàu.



**Hình 5-7:** Một vài ảnh đã qua làm giàu

### 5.3 Cài đặt thí nghiệm

#### 5.3.1 Các mô hình phát hiện biến số

Đề tài thực hiện thí nghiệm trên hai mô hình chính là SSD-ResNet-34 và SSD-ResNet-50 đã được mô tả trong phần mô hình đề xuất. Cả hai mạng đều được huấn luyện trên tập

dữ liệu thật và dữ liệu nhân tạo. Các thí nghiệm chủ yếu là để đánh giá về hiệu quả phát hiện, tốc độ chạy và ảnh hưởng của dữ liệu trên mô hình. Cấu hình để thực hiện huấn luyện hai mạng này là như nhau:

- Kích thước bộ (batch size): 8
- Số lớp đối tượng: 3 (2 lớp biển số và 1 lớp nền)
- Số epoch: 200
- Learning rate: 0.001
- Momentum: 0.5
- Chu kỳ giảm learning rate: [70, 120, 150]
- Hệ số giảm learning rate: 0.5

### 5.3.2 Các mô hình nhận diện biển số

Trên cơ sở mạng CRNN [1] đề tài thực hiện các thí nghiệm với dữ liệu thực và dữ liệu nhân tạo. Bên cạnh đó nhận thấy góc xoay và độ biến dạng của biển số có ảnh hưởng lớn đến quá trình huấn luyện cũng như dự đoán. Do đó đề tài thực hiện một vài thí nghiệm trên ảnh được biến đổi xoay hoặc phôi cảnh sao cho biển số được canh thẳng nhất có thể. Để có được các ảnh này, đề tài dựa trên dữ liệu 4 góc của biển số đã được gán trước đó và ước lượng góc xoay cũng như ma trận biến đổi phôi cảnh để biến đổi ảnh biển số đã cắt. Từ đó đề xuất kiến trúc mạng CRNN cải tiến như đã trình bày trong mục như đã trình bày trong mục 4.4. Cấu hình để thực hiện huấn luyện hai mạng này là tương tự nhau:

- Kích thước bộ (batch size): 8
- Số lớp đối tượng: 23 (số lượng ký tự)
- Số epoch: 70
- Learning rate: 0.01

- Momentum: 0.5
- Chu kỳ giảm learning rate: [40, 50, 60]
- Hệ số giảm learning rate: 0.5

### 5.3.3 Cấu hình hệ thống thí nghiệm

Các thí nghiệm được thực hiện trên máy tính có cấu hình phần cứng như sau:

- CPU Intel Core i7 2.50GHz.
- 16GB RAM.
- GPU GTX 1080.

Về mặt phần mềm, môi trường để chạy các thí nghiệm là hệ điều hành Ubuntu16.04. Các thí nghiệm hiện thực bằng ngôn ngữ lập trình python, sử dụng thư viện học sâu chính là PyTorch phiên bản 0.3.1 và thư viện hàm lỗi CTC của phòng nghiên cứu Baidu. Thời gian huấn luyện các thí nghiệm trên máy được thể hiện như bảng sau:

Mô hình	Thời gian huấn luyện
SSD-ResNet-34	12 giờ
SSD-ResNet-50	16 giờ
CRNN	15 phút
CRNN-STN	15 phút

Bảng 5-1: Thời gian huấn luyện các mô hình trên máy

## 5.4 Kết quả thí nghiệm

### 5.4.1 Các mô hình phát hiện biển số

Các mô hình về phát hiện biển số xe nhìn chung không có sự chênh lệch lớn về độ chính xác. Tuy nhiên vẫn còn tồn tại một số trường hợp mà tại đó biển số xe vẫn bị nhầm với các biển báo, hàng rào và nhựa đường.

Các trường hợp nhận nhầm biển số xe thường là những vùng ảnh có cấu trúc tương tự với vùng biển số xe như màu sắc hoặc có các ký tự trên đó. Để giải quyết vấn đề này, người dùng để xác định một vùng có phải là biển số hay không được tăng lên từ 0.5 lên 0.7 sẽ giúp giảm bớt việc phát hiện nhầm. Hơn nữa sau khi qua bước nhận diện nếu vùng biển số không có nhận được ký tự nào cũng sẽ được bỏ qua. Một số ảnh bị nhận diện sai vùng biển số:



**Hình 5-8:** Vùng đèn xe bị nhầm với biển số

Đề tài cũng thử nghiệm huấn luyện hai mạng trên hoàn toàn với dữ liệu nhân tạo. Sau đó đánh giá trên dữ liệu thật. Kết quả về độ chính xác tuy thấp hơn so với dữ liệu thật nhưng vẫn trong mức chấp nhận được. Do đó việc sử dụng dữ liệu nhân tạo để huấn luyện mạng phát hiện là hoàn toàn khả thi.

**Bảng 5-2:** Kết quả phát hiện biển số xe

Mô hình	Dữ liệu huấn luyện	Dữ liệu kiểm thử	mAP
SSD-ResNet-34	Thực tế	Thực tế	91.92%
SSD-ResNet-50	Thực tế	Thực tế	91.8%
SSD-ResNet-34	Nhân tạo	Thực tế	90.17%
SSD-ResNet-50	Nhân tạo	Thực tế	90.25%

Với sự thay đổi không đáng kể trong độ chính xác, việc thay thế mô hình SSD-ResNet-50 thành mô hình SSD-ResNet-34 nhằm tăng tốc cho quá trình nhận diện là phù hợp. Hơn nữa với kiến trúc nhỏ hơn mạng SSD-ResNet-34 còn giúp giảm lượng bộ nhớ sử dụng, từ đó có thể tăng kích thước bó lén nhiều lần và làm giảm thời gian huấn luyện xuống. Từ thực nghiệm, với hệ thống máy cấu hình như trên, kích thước bó của mạng SSD-ResNet-34 có thể lên đến 16. Bảng dưới đây so sánh tốc độ chạy giữa 2 mô hình phát hiện:

**Bảng 5-3:** Tốc độ chạy của hai mô hình mạng phát hiện

Mô hình	Kích thước batch Batch size	Tốc độ FPS
SSD-ResNet-34	8	60
SSD-ResNet-50	8	35.91

#### 5.4.2 Các mô hình nhận diện biển số

Để xác định được hướng cải tiến, đề tài đã thực hiện đánh giá trên dữ liệu ảnh đã qua các phép biến đổi xoay và phôi cảnh như đã mô tả trong mục 5.3.2. Kết quả được trình bày như bảng cho thấy việc căn chỉnh ảnh đầu vào có ảnh hưởng rõ rệt đến việc nhận dạng chuỗi ký tự. Độ chính xác của mô hình CRNN [1] gốc được cải thiện từ 78.02% lên 85.76%.

**Bảng 5-4:** Kết quả trên tập dữ liệu trải qua phép biến đổi

Phép biến đổi	Độ chính xác
Không dùng	78.02%
Xoay	79%
Phối cảnh	85.76%

Đề tài thực hiện đánh giá mô hình CRNN [1] gốc cùng với mô hình đề xuất CRNN-STN trên cả hai tập dữ liệu thật và dữ liệu nhân tạo cho kết quả như bảng sau:

**Bảng 5-5:** Kết quả trên tập dữ liệu thực tế và nhân tạo

Mô hình	Dữ liệu huấn luyện	Dữ liệu kiểm thử	Độ chính xác
CRNN [1]	Thực tế	Thực tế	78.02%
CRNN-STN	Thực tế	Thực tế	81.73%
CRNN [1]	Thực tế + Nhân tạo (1:1)	Thực tế	80.91%
CRNN-STN	Thực tế + Nhân tạo (1:1)	Thực tế	82.83%

Từ kết quả thí nghiệm với ảnh qua căn chỉnh, đề tài hiện thực mô hình CRNN cải tiến. Kết quả đánh giá trên tập dữ liệu thật cho thấy độ chính xác được cải thiện lên thành 82.83% so với mô hình gốc. Ngoài ra từ bảng kết quả trên cũng cho thấy việc bổ sung dữ liệu nhân tạo trong quá trình huấn luyện cũng góp phần cải thiện độ chính xác của mô hình gốc lẫn mô hình đề xuất. Tuy nhiên dữ liệu nhân tạo vẫn còn khá dễ so với tập dữ liệu thật do các ký tự vẫn còn khá rõ nét. Do đó việc sử dụng hoàn toàn biến số sinh tự động để huấn luyện mạng nhận diện chưa thể áp dụng vào điều kiện thực tế.

Một số kết quả của mô hình phát hiện và nhận diện biến số như trong hình 5-9. Có thể thấy mô hình phát hiện và nhận diện tương đối tốt cho đa phần các trường hợp. Tuy nhiên

mô hình vẫn còn những sai sót như nhận diện thiếu ký tự hoặc sai trong một số trường hợp khó:



**Hình 5-9:** Kết quả phát hiện và nhận dạng biển số

## CHƯƠNG 6: TỔNG KẾT

Đề tài đã xây dựng một mô hình tích hợp giúp phát hiện và nhận dạng biển số xe cùng lúc. Kết quả đạt được tương đối khả quan và có thể được ứng dụng trong thực tế. Ưu điểm của mô hình là không cần trải qua giai đoạn phân đoạn ký tự, giúp giảm bớt các bước hậu xử lý cũng như sai số từ bước này gây ra. Việc giảm kích thước mạng phát hiện cũng giúp giảm đáng kể thời gian huấn luyện cũng như tăng tốc cho quá trình chạy thực tế. Tuy nhiên độ chính xác của việc nhận diện còn chưa cao. Bên cạnh đó dữ liệu dùng cho việc huấn luyện vẫn còn hạn chế về mặt số lượng dẫn đến chưa đánh giá hết khả năng của mô hình.

Trong tương lai, đề tài sẽ hướng tới việc cải thiện độ chính xác của việc nhận dạng ký tự trong biển số xe. Trong khi bước phát hiện có độ chính xác khá cao, thì bước nhận dạng vẫn là thử thách khá lớn. Bên cạnh đó đề tài cũng sẽ hướng tới việc xử lý trực tiếp từ video hoặc camera để mô hình có thể ứng dụng được trong các hệ thống phân tích thời gian thực.

# TÀI LIỆU THAM KHẢO

- [1] 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 transactions on pattern analysis and machine intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.
- [2] Z. Xu, W. Yang, A. Meng, N. Lu, H. Huang, C. Ying, and L. Huang, “Towards end-to-end license plate detection and recognition: A large dataset and baseline,” in *European Conference on Computer Vision*, pp. 261–277, Springer, 2018.
- [3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*, pp. 21–37, Springer, 2016.
- [4] T. K. Cheang, Y. S. Chong, and Y. H. Tay, “Segmentation-free vehicle license plate recognition using convnet-rnn,” *arXiv preprint arXiv:1701.06439*, 2017.
- [5] H. Li, P. Wang, and C. Shen, “Towards end-to-end car license plates detection and recognition with deep neural networks,” *arXiv preprint arXiv:1709.08828*, 2017.
- [6] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *arXiv preprint arXiv:1207.0580*, 2012.
- [7] Y. LeCun *et al.*, “Lenet-5, convolutional neural networks,” *URL: http://yann. lecun. com/exdb/lenet*, p. 20, 2015.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

- [9] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, pp. 91–99, 2015.
- [10] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, “S<sup>^</sup> 3fd: Single shot scale-invariant face detector,” in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 192–201, IEEE, 2017.
- [11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.
- [12] S. Rasheed, A. Naeem, and O. Ishaq, “Automated number plate recognition using hough lines and template matching,” in *Proceedings of the World Congress on Engineering and Computer Science*, vol. 1, pp. 24–26, 2012.
- [13] K. K. Kim, K. Kim, J. Kim, and H. J. Kim, “Learning-based approach for license plate recognition,” in *Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop*, vol. 2, pp. 614–623, IEEE, 2000.
- [14] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [15] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [16] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, ACM, 2006.

- [17] S. Z. Masood, G. Shu, A. Dehghan, and E. G. Ortiz, “License plate detection and recognition using deeply learned convolutional neural networks,” *arXiv preprint arXiv:1703.07330*, 2017.
- [18] H. Li and C. Shen, “Reading car license plates using deep convolutional neural networks and lstms,” *arXiv preprint arXiv:1601.05610*, 2016.
- [19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- [21] C. Bartz, H. Yang, and C. Meinel, “Stn-ocr: A single neural network for text detection and text recognition,” *arXiv preprint arXiv:1707.08831*, 2017.
- [22] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu, “Spatial transformer networks,” in *Advances in Neural Information Processing Systems 28* (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds.), pp. 2017–2025, Curran Associates, Inc., 2015.
- [23] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, “Icdar 2003 robust reading competitions,” in *Proc. Seventh Int. Conf. Document Analysis and Recognition*, pp. 682–687, Aug. 2003.
- [24] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti, “A robust real-time automatic license plate recognition based on the yolo detector,” in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, July 2018.

- [25] G.-S. Hsu, J.-C. Chen, and Y.-Z. Chung, “Application-oriented license plate recognition,” *IEEE transactions on vehicular technology*, vol. 62, no. 2, pp. 552–561, 2013.
- [26] Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu, and N. Komodakis, “A robust and efficient approach to license plate detection,” *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1102–1114, 2017.

# LÝ LỊCH TRÍCH NGANG

Họ và tên: Nguyễn Xuân Thành

Ngày, tháng, năm sinh: 21/06/1992

Nơi sinh: Bình Dương

Địa chỉ liên lạc: 9/33, Bình Đường I, An Bình, Dĩ An, Bình Dương.

## QUÁ TRÌNH ĐÀO TẠO

Thời gian	Trường
2010-2015	Đại học Bách Khoa TP. HCM (Sinh viên đại học)
2016-2018	Đại học Bách Khoa TP. HCM (Học viên cao học)

## QUÁ TRÌNH CÔNG TÁC

Thời gian	Nơi công tác	Chức vụ
2015-2018	Công ty FPT Software	Lập trình viên
2018	Bộ môn khoa học máy tính	Trợ giảng
2018-hiện tại	Phòng thí nghiệm GVLab	Nghiên cứu viên