

VIỆN NGHIÊN CỨU DỮ LIỆU LỚN



**NHẬN DẠNG KÝ TỰ QUANG HỌC
ỨNG DỤNG ĐỌC BIẾN SỐ XE VIỆT NAM**

Giảng viên hướng dẫn: TS Nguyễn Quang Uy

Thành viên:

Hoàng Công Đức (Nhóm trưởng)

Trần Tiến Hiệp

Đỗ Minh Hiếu

Nguyễn Thị Minh Thu

Mục lục

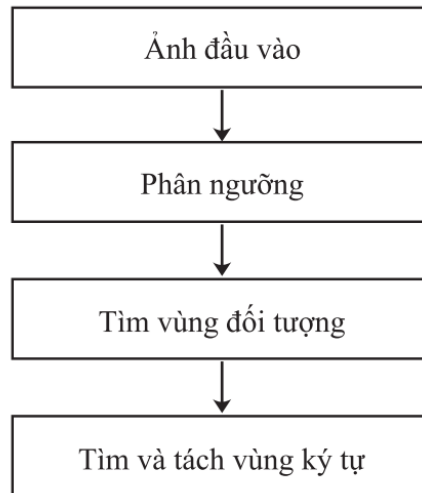
1. Giới thiệu về đề tài	3
2. Mục tiêu	4
3. Ý tưởng thực hiện	5
4. Dữ liệu	6
5. Phát hiện vùng ký tự trong ảnh	8
6. Phân loại kí tự	11
a. Mô hình SVM	11
b. Mô hình CNN sử dụng Keras	11
7. Đánh giá kết quả	13
a. Độ chính xác của từng mô hình huấn luyện	13
b. Phương pháp đánh giá nhận diện biển số xe	14
c. Điểm yếu của mô hình	14
8. Triển khai mô hình	15
9. Kết luận	16

1. Giới thiệu về đề tài

Nhận dạng biển số tự động là công nghệ thị giác máy tính giúp nhận dạng biển số xe từ hình ảnh một cách hiệu quả mà không cần sự can thiệp của con người. Trong những năm gần đây, nó ngày càng trở nên quan trọng hơn do ba yếu tố chính: số lượng phương tiện tham gia giao thông ngày càng tăng, sự phát triển nhanh chóng của kỹ thuật xử lý hình ảnh và khả năng ứng dụng thực tế mà công nghệ này mang lại. Một số ứng dụng điển hình nhất của hệ thống ANPR là giám sát giao thông, thu phí tự động hoặc kiểm soát ra vào bãi đậu xe. Ngoài ra, công nghệ này cũng được sử dụng rộng rãi cho các mục đích khác như an ninh trật tự, giúp xác định xe của tội phạm.

Tuy nhiên, việc phát triển hệ thống ANPR không phải là nhiệm vụ dễ dàng, vì nó phải đối mặt với nhiều thách thức do môi trường và sự biến dạng của biển số. Các kiểu chiếu sáng hoặc nền khác nhau ảnh hưởng rất nhiều đến việc nhận dạng biển số vì nó có thể làm giảm chất lượng của hình ảnh và các mẫu nền gây thêm khó khăn cho quá trình xác định vị trí biển số. Đối với vấn đề thứ hai, vị trí, số lượng, kích thước, phong chữ, màu sắc hoặc độ nghiêng của biển số tạo thành những yếu tố rất thách thức trong việc phát triển một hệ thống ANPR nhất quán.

2. Mục tiêu



Mục tiêu của dự án này là có thể phát triển một hệ thống tự động nhận dạng biển số xe máy Việt Nam từ hình ảnh kỹ thuật số bằng cách sử dụng một số thuật toán học máy. Do đó, các mục tiêu quan trọng nhất trong dự án là:

- Phân tích các phương pháp Học máy khác nhau và chọn những phương pháp phù hợp nhất cho ANPR, tìm hiểu cách chúng hoạt động và điều chỉnh chúng phù hợp với ứng dụng ANPR.
- Phát triển một ứng dụng ANPR hiệu quả, nhanh chóng và đáng tin cậy có thể nhận diện được biển số xe.
- Tạo tập dữ liệu hình ảnh biển số xe đầy đủ để có thể vừa huấn luyện, vừa kiểm tra các thuật toán học máy và đánh giá hiệu suất của hệ thống đã phát triển.

Trong quy mô dự án, chúng tôi giới hạn bài toán nhận diện biển số xe từ ảnh đã được cắt chỉ còn biển số (không có xe). Trong hình ảnh chỉ chứa một biển số xe duy nhất. Đầu vào là ảnh chụp phương tiện cùng với biển số. Đầu ra là các ký tự trên biển số xe dưới dạng text.

3. Ý tưởng thực hiện

Sử dụng thư viện OpenCV là một thư viện mã nguồn mở chuyên xử lý về hình ảnh kết hợp với các thuật toán học máy cơ bản

Dùng xử lý ảnh để phát hiện vùng chứa kí tự trong ảnh.

Đây là một bước quan trọng trong quy trình nhận dạng biển số xe tự động bởi vì tất cả các bước sau đó đều phụ thuộc vào bước này. Nếu tách ký tự không chính xác thì mọi cố gắng tiếp theo đều vô nghĩa. Việc tách ký tự có thể gặp phải một số khó khăn như dính ký tự, dính viền biển số với ký tự, không thể tách được ký tự do ký tự quá mờ hoặc biển số quá cũ. Đôi khi ký tự tách được lại có một số nhiễu đi kèm. Đặc biệt đối với Việt Nam, rất nhiều ký tự trên biển số bị che lấp bởi các đinh ốc gắn vào biển. Đây là một thách thức lớn đối với quá trình nhận dạng. Có rất nhiều phương pháp để thực hiện tách ký tự, một số phương pháp có thể kể đến như phương pháp chiếu, phương pháp hình thái học, phương pháp phát hiện contour, phương pháp sử dụng trí tuệ nhân tạo... Trong bài báo cáo này chúng tôi sẽ tập trung vào phương pháp contour.

Nhận dạng ký tự dựa trên phương pháp SVM và CNN

Sau khi tách rời được các ký tự, việc tiếp theo là nhận dạng ký tự. Bài toán nhận dạng ký tự là bài toán đã được phát triển và sử dụng trong rất nhiều ứng dụng như nhận dạng chữ viết tay, nhận dạng chữ in. Với bài toán nhận dạng biển số xe, số lượng ký tự ít, các kiểu font chữ ít cho nên độ phức tạp cũng ít hơn, đó là một thuận lợi. Phương pháp nhận dạng ký tự phổ biến là sử dụng mạng nơ-ron tức là huấn luyện cho máy tính nhận dạng các ký tự. Phương pháp này cho kết quả khá chính xác.

Trong bài báo cáo này mà chúng tôi lựa chọn là phương pháp nhận dạng SVM (support vector machine). Phương pháp SVM sử dụng một không gian đa chiều để phân hoạch các mẫu. Các mẫu nhận dạng được biểu diễn dưới dạng véc tơ. Ban đầu, thuật toán dựa vào tập mẫu để phân hoạch không gian véc tơ thành các không gian con. Mỗi không gian con tương ứng với một loại mẫu. Dựa vào các không gian con này, phương pháp SVM có thể nhận dạng được các mẫu. Phương pháp này sẽ được trình bày cụ thể ở những phần sau.

4. Dữ liệu

Cần chuẩn bị cơ sở dữ liệu là tập hợp các ký tự trong biển số xe. Có 31 ký tự thường gặp trong biển số xe, do đó cần phân loại 31 lớp này

Dữ liệu được sử dụng bao gồm:

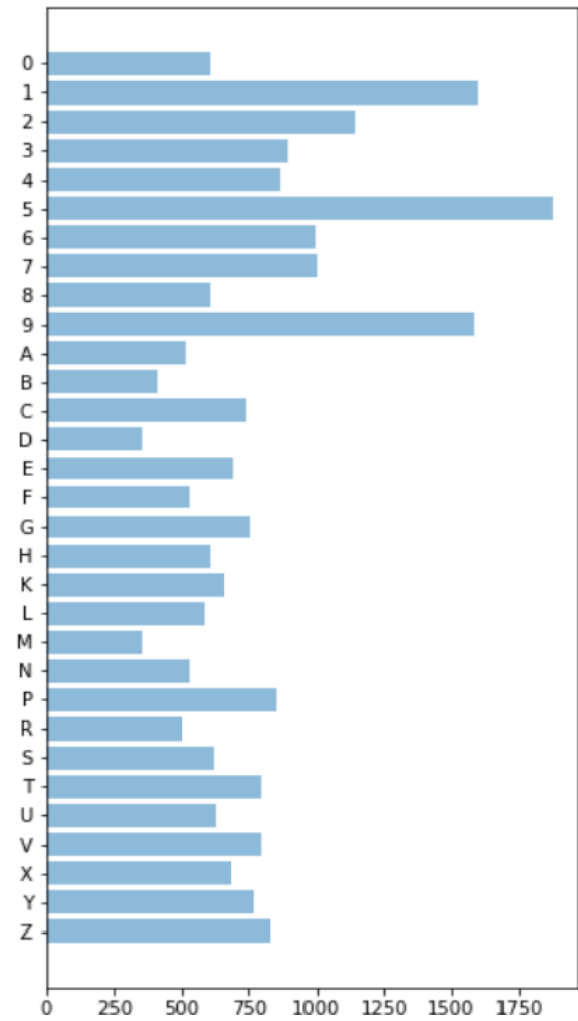
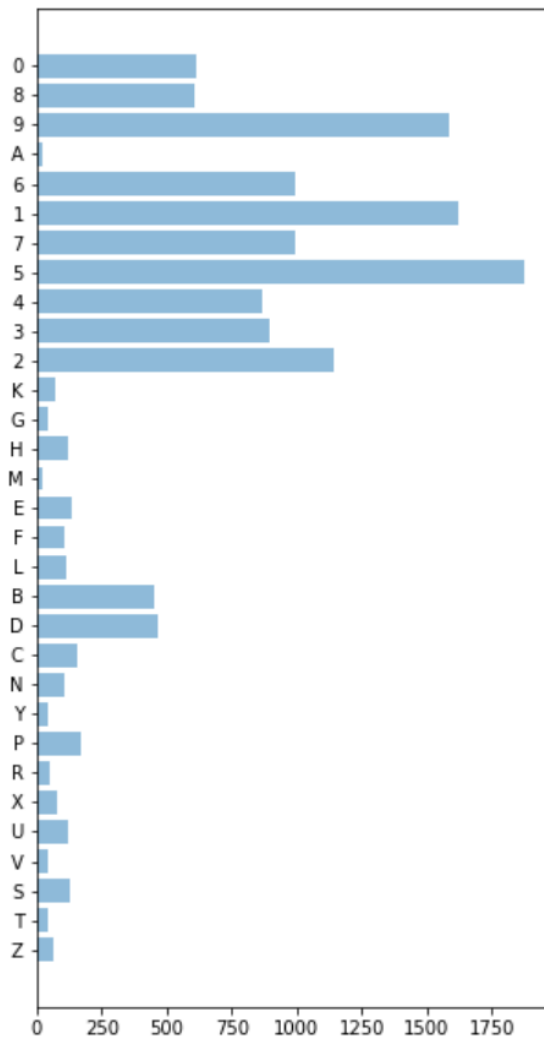
- Bộ ảnh biển số xe máy của cty GreenParking
- Bộ ảnh biển số xe tự chụp bằng điện thoại di động

Bộ dữ liệu bao gồm 1855 biển số xe được gán nhãn thủ công nhằm giảm thiểu tối đa số lượng dữ liệu bị gán nhãn không chính xác.

Sau khi thu thập được dữ liệu hình ảnh, chúng tôi tiến hành làm sạch dữ liệu bằng cách cắt đi những phần hình ảnh xung quanh biển số xe để thu về hình ảnh chỉ chứa một biển số xe duy nhất.

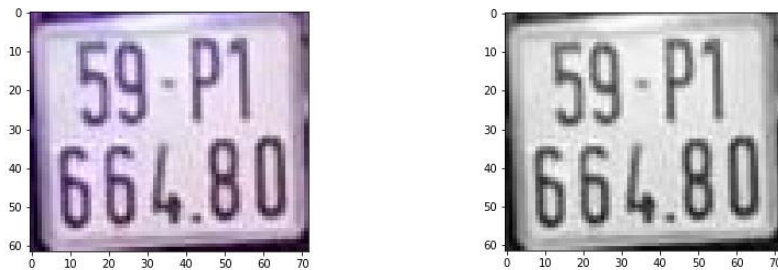
Các ký tự trong biển số xe được cắt ra và phân loại cho mục đích huấn luyện từng ký tự, thu được tổng cộng hơn 13 nghìn ảnh từng ký tự có phân bố không cân bằng giữa phần ký tự số (10 số có tổng cộng 11160 ảnh) và phần ký tự chữ (21 chữ có tổng cộng 1852 ảnh). Để giải quyết vấn đề mất cân bằng dữ liệu, chúng tôi tạo thêm các ảnh ký tự chữ bằng cách sử dụng các hàm tạo nhiễu `random_noise` của thư viện `skimage` và mở rộng, thu hẹp nét `dilate` và `erode` của `opencv` trên ảnh gốc để thu được ảnh mới.

Sau đó, dữ liệu được chia ra làm 2 tập train và test. Tập train khoảng 90% tập dữ liệu (hơn 20 nghìn ảnh), tập test chứa phần còn lại (khoảng 2.3 nghìn ảnh).

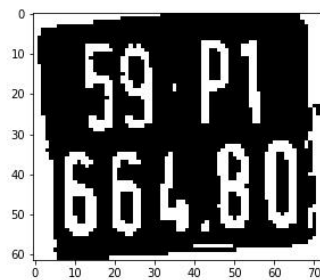


5. Phát hiện vùng ký tự trong ảnh

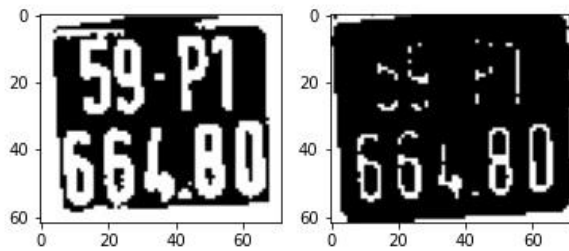
Để phát hiện các vùng chứa ký tự trong ảnh, ta sử dụng phương pháp dựa trên việc xử lý ảnh tìm ra các mảng màu đen trên nền biển số màu trắng. Với cách tiếp cận này, ta không cần quan tâm nhiều đến màu sắc cụ thể của các điểm ảnh, mà chỉ cần quan tâm tới độ sáng, nên ở bước đầu tiên, ảnh sẽ được chuyển sang ảnh đen trắng (grayscale).



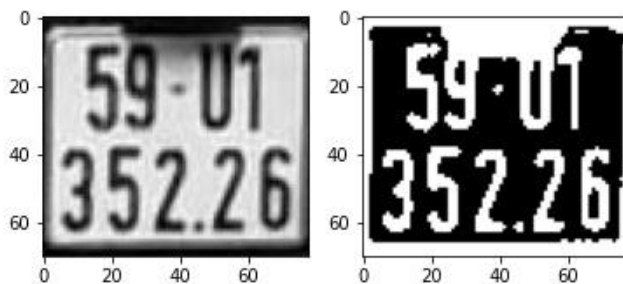
Để có thể tìm các contour, ảnh cần được đưa về dạng ảnh nhị phân. Một giá trị ngưỡng sẽ được chọn, những điểm ảnh có giá trị thấp hơn được đưa về 0 (màu đen), những điểm còn lại đưa về 255 (màu trắng). Tuy nhiên, do những vùng số là vùng cần phát hiện, ta làm ngược lại, các vùng dưới ngưỡng sẽ thành màu trắng và các vùng trên ngưỡng sẽ thành màu đen.



Để thuật toán hoạt động hiệu quả, ngưỡng giá trị cần được lựa chọn phù hợp. Nếu ngưỡng quá cao, sẽ có nhiều điểm trên nền biển số bị nhầm thành khu vực chứa ký tự (màu đen) trong khi nếu ngưỡng quá thấp, những vùng ký tự bị chiếu sáng có thể không được phát hiện ra.

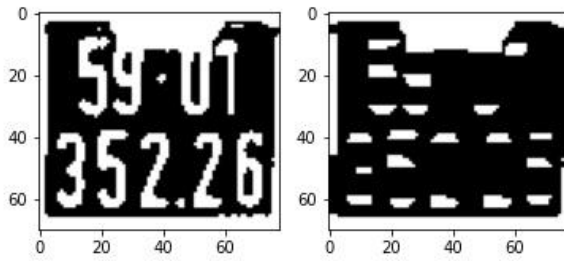


Để chọn ngưỡng phù hợp, ta sử dụng phương pháp chọn ngưỡng Otsu. Tuy nhiên, trong một số trường hợp, do điều kiện ánh sáng không tốt, phương pháp này cũng không thể giúp tìm ra một giá trị ngưỡng tốt, khiến cho ảnh nhị phân thu được không được như mong muốn.

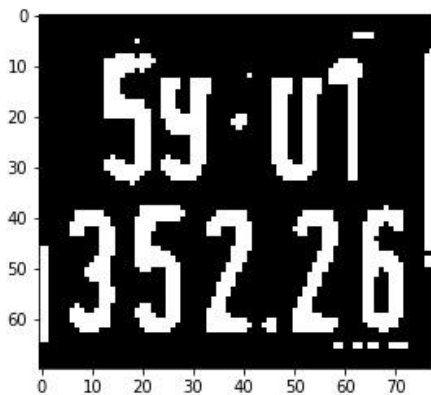


Để tìm ra các vùng chứa ký tự từ ảnh nhị phân, ta sẽ tìm các contour bao quanh các vùng trắng. Tuy nhiên, do các vùng trắng này có thể bị dính vào phần rìa ngoài, khiến cho các contour tìm được không chính xác, ta tìm cách loại bỏ phần rìa ngoài của biển số trước khi thực hiện tìm contour.

Ta thực hiện bước này như sau. Trước tiên, ta sử dụng phép biến đổi morphological transformation, cụ thể là phép open, để tạo ra đường ngăn cách giữa phần rìa ngoài và các ký tự số.



Sau đó, trên ảnh thu được, ta sẽ tìm những vùng màu trắng nằm sát cạnh của khung hình. Những vùng đó là vùng nằm ngoài biển số trên khung hình. Dựa vào thông tin về vị trí của những vùng này, ta sẽ tô đen những vùng tương ứng trên ảnh nhị phân ban đầu. Khi đó, phần lớn những điểm ảnh trắng tương ứng với vùng ngoài biển số sẽ được đưa về màu đen.



Đến đây, ta có thể tìm các contour bao quanh các khu vực màu trắng trên ảnh. Đây là những khu vực có khả năng cao là vùng chứa kí tự. Từ mỗi contour này, ta sẽ tìm một hình chữ nhật nhỏ nhất chứa contour đó.

Trên ảnh sau khi đã tô đen phần ngoài biển số, ngoài những vùng chứa kí tự, còn có một số vùng trắng nhỏ xuất hiện do ánh sáng của ảnh chụp không rõ hoặc do các bước xử lý trước sinh ra. Dựa vào kích thước của các hình chữ nhật ở trên, ta có thể xác định được vùng ảnh nằm trong hình chữ nhật nào là vùng chứa kí tự của biển số.

6. Phân loại ký tự

a. Mô hình SVM

Thuật toán trích xuất đặc trưng: SIFT (Scale-invariant feature transform) là một feature descriptor được sử dụng trong computer vision và xử lý hình ảnh được dùng để nhận dạng đối tượng, matching image, hay áp dụng cho các bài toán phân loại...

Mô hình cho nhận dạng chữ cái và chữ số: SVM là một thuật toán phân loại dữ liệu, dữ liệu đối với các ký tự mà ta cần nhận dạng ở đây chính là các đặc trưng trong ảnh của ký tự đó..

Tối ưu siêu tham số cho mô hình học máy: GridSearchCV cho phép kết hợp một công cụ ước tính với phần mở đầu tìm kiếm lưới để điều chỉnh các tham số siêu. Phương thức chọn tham số tối ưu từ tìm kiếm lưới và sử dụng nó với công cụ ước tính do người dùng chọn. GridSearchCV kế thừa các phương thức từ trình phân loại, vì vậy, có thể sử dụng các phương thức `.score`, `.predict`, v.v. trực tiếp thông qua giao diện GridSearchCV. Nếu muốn trích xuất các tham số siêu tốt nhất được xác định bởi tìm kiếm dạng lưới, ta có thể sử dụng `.best_params_` và điều này sẽ trả về siêu tham số tốt nhất. Sau đó, ta có thể chuyển siêu tham số này cho công cụ ước tính của mình một cách riêng biệt.

b. Mô hình CNN sử dụng Keras

Chuẩn hóa đầu vào: Ảnh từng ký tự thu được từ bước xử lý trước đó được resize về kích thước 32x32 (chiều cao x chiều rộng). Sau đó chuyển ảnh về dạng nhị phân để model dễ tính toán.

Cấu trúc neural network: sử dụng một model CNN đơn giản để tiến hành trích xuất đặc trưng và phân loại. Model này được thiết kế với các số filters tăng dần [32, 64, 64] để càng gần về output không gian đặc trưng giảm nhưng số lượng cách học tăng dần.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 32, 16, 32)	320
conv2d_1 (Conv2D)	(None, 30, 14, 32)	9248
max_pooling2d (MaxPooling2D)	(None, 15, 7, 32)	0
dropout (Dropout)	(None, 15, 7, 32)	0
conv2d_2 (Conv2D)	(None, 15, 7, 64)	18496
conv2d_3 (Conv2D)	(None, 13, 5, 64)	36928
max_pooling2d_1 (MaxPooling2D)	(None, 7, 3, 64)	0
dropout_1 (Dropout)	(None, 7, 3, 64)	0
conv2d_4 (Conv2D)	(None, 7, 3, 64)	36928
conv2d_5 (Conv2D)	(None, 5, 1, 64)	36928
max_pooling2d_2 (MaxPooling2D)	(None, 3, 1, 64)	0
dropout_2 (Dropout)	(None, 3, 1, 64)	0
flatten (Flatten)	(None, 192)	0
dense (Dense)	(None, 512)	98816
dropout_3 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 31)	15903
Total params: 253,567		
Trainable params: 253,567		
Non-trainable params: 0		

Huấn luyện mô hình: các tham số huấn luyện

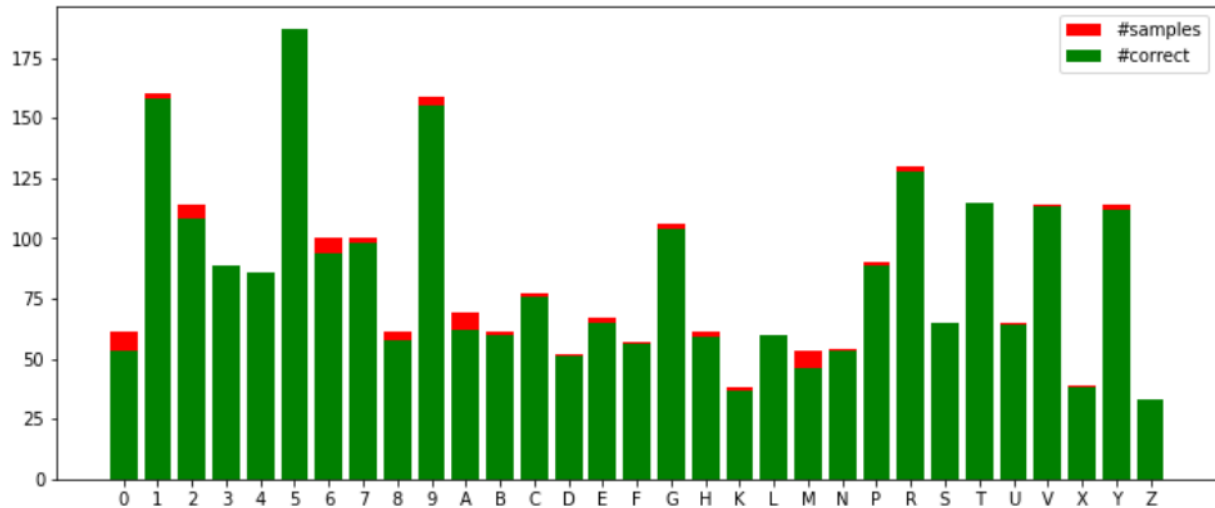
Hàm loss: categorical cross entropy

Hàm tối ưu: Adam với tốc độ học 1e-5

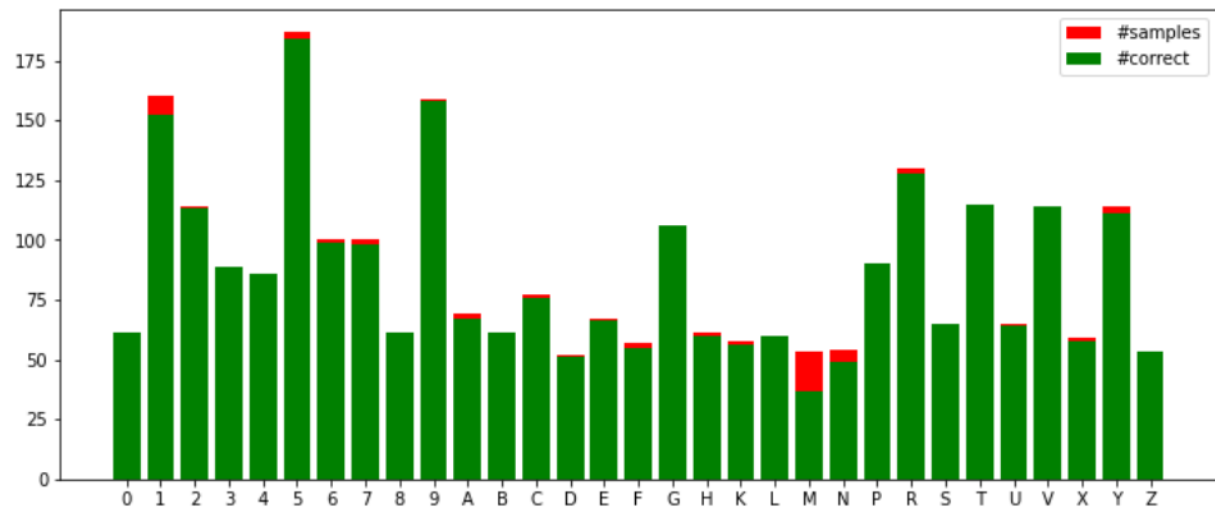
7. Đánh giá kết quả

a. Độ chính xác của từng mô hình huấn luyện

SVM



CNN:



Nhìn vào biểu đồ độ chính xác của model trên từng ký tự nhận thấy mô hình tương đối tốt trên hầu hết các ký tự, có độ chính xác trung bình trên tập test xấp xỉ 98.483%.

Chỉ có ký tự chữ “M” kém hơn hẳn do tập dữ liệu thu được để huấn luyện có hình ảnh chữ M rất khó phân biệt với các chữ N và H kể cả bằng mắt thường.



b. Phương pháp đánh giá nhận diện biển số xe

Chúng tôi sử dụng khoảng cách Levenshtein để đánh giá độ hiệu quả, hiệu suất của hệ thống ANPR. Khoảng cách Levenshtein thể hiện khoảng cách khác biệt giữa 2 chuỗi ký tự. Khoảng cách Levenshtein giữa chuỗi S và chuỗi T là số bước ít nhất biến chuỗi S thành chuỗi T thông qua 3 phép biến đổi là xóa 1 ký tự, thêm 1 ký tự và thay ký tự này bằng ký tự khác. Khoảng cách Levenshtein càng nhỏ chứng tỏ hệ thống nhận diện biển số xe càng hiệu quả.

Khoảng cách Levenshtein trung bình của các phương pháp CNN và SVM lần lượt là 257/110 và 259/110 kiểm tra trên tập dữ liệu bao gồm 110 ảnh biển số xe tự chụp ngẫu nhiên.

c. Điểm yếu của mô hình

Mô hình hoạt động kém hiệu quả khi chất lượng ảnh chụp biển số bị mờ, ánh sáng kém, độ sáng các vùng ảnh quá chênh lệch dẫn đến việc nhận diện vùng chứa ký tự xảy ra sai sót. Từ đó, việc mất hay nhận diện nhầm vùng chứa ký tự là điều không thể tránh khỏi làm cho hệ thống nhận diện sai biển số xe.

8. Triển khai mô hình

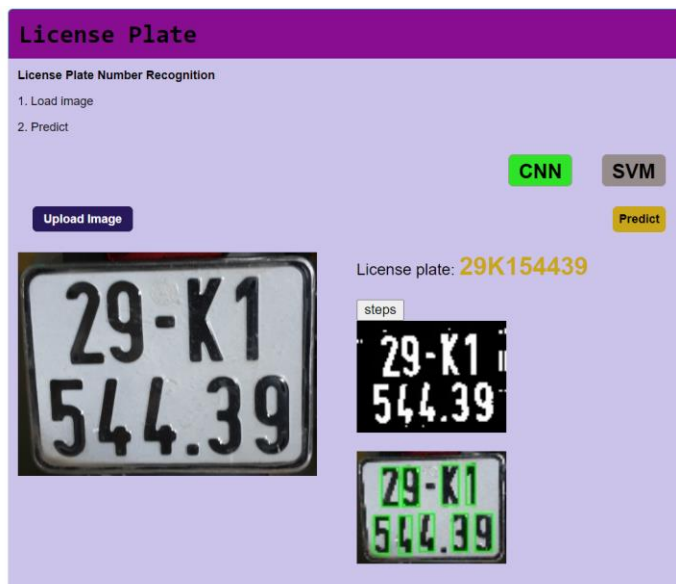
Giao diện chương trình thử nghiệm



Bao gồm:

- Nút “Upload image”: Mở ảnh biển số xe
- Nút “CNN” và “SVM”: nhận dạng
- Nhãn “License plate”: hiển thị kết quả biển số
- Nút “steps”: hiển thị thêm ảnh ở bước tiền xử lí

Khi click vào nút “steps” sẽ có kết quả sau:



9. Kết luận

Dự án Nhận diện ký tự quang học, nhận diện biển số xe đã nghiên cứu và thử nghiệm các phương pháp học máy cơ bản là SVM và CNN cùng với các kỹ thuật xử lý hình ảnh trước khi đưa vào mô hình huấn luyện sử dụng thư viện mã nguồn mở OpenCV. Hai mô hình học máy SVM và CNN đã được sử dụng và trả về kết quả khá tốt cho các đầu vào là ảnh biển số xe. Qua kiểm nghiệm trên tập dữ liệu thực tế chưa qua huấn luyện, chúng tôi nhận thấy rằng mô hình CNN đã huấn luyện cho kết quả tốt nhất. Chúng tôi nhận thấy rằng các phương pháp tiền xử lý ảnh đóng vai trò rất quan trọng trong việc dự đoán kết quả đầu ra của mô hình.

Với kết quả thực nghiệm trên, hướng nghiên cứu phát triển tiếp theo của em sẽ là:

- Nâng cao hiệu quả chương trình nhận dạng ký tự, cách ly các ký tự trong biển số trong các trường hợp biển số bị nhiễu nhiễu, mất mát thông tin do nhiễu từ điều kiện môi trường
- Mở rộng tập ảnh mẫu để tăng độ chính xác nhận dạng
- Mở rộng điều kiện nhận dạng với những biển số bị nghiêng nhiễu, biển số nền xanh chữ trắng, nền đỏ chữ trắng. Và tiến tới nhận dạng nhiễu nhiễu loại biển nước ngoài.
- Phát triển module nhận dạng vùng biển số xe để tích hợp với module này thành một ứng dụng có thể triển khai và có tính khả dụng cao

REFERENCES

- [1] In-Depth: Support Vector Machines <https://jakevdp.github.io/PythonDataScienceHandbook/05.07-support-vector-machines.html>
- [2] Cristianini, Nello; Shawe-Taylor, John (2000). An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press. ISBN 0-521-78019-5.
- [3] Steinwart, Ingo; Christmann, Andreas (2008). Support Vector Machines. New York: Springer. ISBN 978-0-387-77241-7.
- [4] Adrian Kaehler; Gary Bradski (14 December 2016). Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. O'Reilly Media. pp. 26ff. ISBN 978-1-4919-3800-3.
- [5] OpenCV change logs: <http://code.opencv.org/projects/opencv/wiki/ChangeLog> Archived 2013-01-15 at the Wayback Machine
- [6] Nhận diện biển số xe Việt Nam
<https://viblo.asia/p/nhan-dien-bien-so-xe-viet-nam-Do754P9L5M6>
- [7] OpenCV: Automatic License/Number Plate Recognition (ANPR) with Python
<https://www.pyimagesearch.com/2020/09/21/opencv-automatic-license-number-plate-recognition-anpr-with-python/>