

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN CUỐI KÌ MÔN XỬ LÝ ẢNH SỐ NÂNG CAO

NHẬN DẠNG BIỂN SỐ XE

Người hướng dẫn: **TS PHẠM VĂN HUY**

Người thực hiện: **HỒNG QUANG VINH – 186005004**

NGUYỄN ĐẠI THỊNH – 186005035

Lớp: 18600531

Khoá: 2018-2020

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN CUỐI KÌ MÔN XỬ LÝ ẢNH SỐ NÂNG CAO

NHẬN DẠNG BIỂN SỐ XE

Người hướng dẫn: **TS PHẠM VĂN HUY**

Người thực hiện: **HỒNG QUANG VINH – 186005004**

NGUYỄN ĐẠI THỊNH – 186005035

Lớp: 18600531

Khoá: 2018-2020

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019

LỜI CẢM ƠN

Nhóm chúng em xin chân thành cảm ơn Thầy Phạm Văn Huy đã giúp đỡ chúng em hoàn thành đồ án. Những hướng dẫn của Thầy giúp chúng em có một nền tảng lý thuyết đủ để có thể ứng dụng và nghiên cứu phát triển đề tài này. Xin chân thành cảm ơn Thầy.

ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của TS Phạm Văn Huy;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày tháng năm

Tác giả

(ký tên và ghi rõ họ tên)

Hồng Quang Vinh

Nguyễn Đại Thịnh

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Ngày nay, với sự phát triển của cuộc Cách mạng Công nghiệp 4.0, tiên phong là sự phát triển của lĩnh vực Trí tuệ nhân tạo, con người đang tiến gần hơn tới mục đích làm cho máy tính mô phỏng được các hành vi của con người. Có thể kể đến một số lĩnh vực như nhận dạng hình ảnh, tiếng nói, chữ viết... Trong đó, Thị Giác Máy Tính đang là lĩnh vực có bước phát triển vượt trội nhất, hiện đã và đang được ứng dụng rộng rãi trong các lĩnh vực khoa học và đời sống.

Việc phát triển lĩnh vực này dựa trên ý tưởng sao chép khả năng thị giác của con người. Mục tiêu là thông qua đối tượng thông tin như hình ảnh, máy tính có thể rút được những thông tin cần thiết, từ đó có thể giải quyết vô số bài toán như: nhận dạng vật thể, chú thích ảnh, v.v...

Mục đích của đề tài này là xây dựng một hệ thống mô phỏng khả năng nhận diện bằng thị giác của con người, cụ thể ở đây là nhận dạng biển số xe. Thông qua tập tin hình ảnh hoặc đoạn video, máy tính có thể xác định vị trí biển số cũng như nhận dạng các kí tự nằm trong biển số đó, từ đó hiển thị thông tin ra bên ngoài. Tuy nhiên, để xây dựng hệ thống như vậy trải qua rất nhiều khó khăn, có thể kể đến như công đoạn xử lý bức ảnh cần phải chính xác để có đầu ra lý tưởng phục vụ cho việc huấn luyện. Ngoài ra, dữ liệu về biển số cũng là một trong những khó khăn hàng đầu, vì các dữ liệu này thường không có sẵn, hay bị nhiễu, không chính xác, v.v...

Các khái niệm cũng như quá trình thực hiện sẽ được trình bày ở phần sau của đồ án này.

MỤC LỤC

LỜI CẢM ƠN	i
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	iii
TÓM TẮT	iv
MỤC LỤC	1
DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ	4
CHƯƠNG 1 – MỞ ĐẦU	6
1.1 Giới thiệu đề tài	6
1.2 Mục tiêu và phạm vi đề tài.....	6
1.3 Cơ sở dữ liệu.....	7
CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT	9
2.1 Khái niệm về ảnh	9
2.2 Tiền xử lý.....	10
2.2.1 Chuyển đổi hệ màu	11
2.2.2 Lọc ảnh và khử nhiễu	12
2.2.3 Cân bằng sáng (Histogram Equalization)	14
2.2.4 Xử lý hình thái - Morphological	15
2.2.4.1 Phương pháp ăn mòn (Erosion).....	17
2.2.4.2 Phương pháp giãn nở (Dilation)	17
2.2.4.3 Phương pháp mở (Opening)	18
2.2.5 Phương pháp lấy ngưỡng.....	19
2.3 Connected Component Analysis (CCA).....	20
2.4 Support Vector Machine	24
2.4.1 Giới thiệu:	24
2.4.2 Các khái niệm cơ bản:.....	24
2.4.2.1 Siêu phẳng phân cách:	24
2.4.2.2 Support Vector:.....	28

2.4.2.3 Biên độ Margin:.....	29
2.4.3 Phân lớp dữ liệu:.....	30
Trường hợp dữ liệu có thể phân chia tuyến tính được:.....	30
2.4.3.2 Trường hợp dữ liệu không thể chia tuyến tính được:	31
CHƯƠNG 3 – MÔ HÌNH GIẢI QUYẾT BÀI TOÁN.....	33
3.1 Giới thiệu	33
3.2 Chi tiết mô hình	33
CHƯƠNG 4 – KẾT QUẢ THỰC NGHIỆM.....	34
4.1 Phương pháp đánh giá	34
4.2 Kết quả thực nghiệm.....	34
CHƯƠNG 5 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	36
5.1 Các hạn chế.....	36
5.2 Những điều đã đạt được.....	36
5.3 Hướng phát triển	36

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CÁC KÝ HIỆU

CÁC CHỮ VIẾT TẮT

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

DANH MỤC HÌNH

Hình 1: Ví dụ về ảnh xám.....	10
Hình 2: Ví dụ ảnh màu trên không gian RGB (Red, Green, Blue).....	10
Hình 3: Ảnh đã chuyển qua ảnh xám.....	11
Hình 4: Minh họa việc nhân ma trận ảnh.....	12
Hình 5: Kết quả sau khi sử dụng bộ lọc Bilateral	13
Hình 6: Ví dụ về histogram.....	14
Hình 7: Kết quả sau khi thực hiện cân bằng sáng Histogram Equalization.....	15
Hình 8: Phần tử cấu trúc có kích thước 5x5	16
Hình 9: “fit” và “hit” của ảnh nhị phân với phần tử cấu trúc s1 và s2.....	16
Hình 10: Ảnh gốc (bên trái) và ảnh sau khi sử dụng phương pháp ăn mòn (bên phải).	17
Hình 11: Ảnh gốc (bên trái) và ảnh sau khi sử dụng phương pháp giãn nở (bên phải).	18
Hình 12: Kết quả sau khi thực hiện phương pháp Opening.....	18
Hình 13: Kết quả sau khi thực hiện phương pháp Closing.....	19
Hình 14: Ảnh sau khi lấy ngưỡng.....	20
Hình 15: Lận cận 4.....	21
Hình 16: Lận cận 8.....	21
Hình 17: Hình ảnh đầu vào	22
Hình 18: Hình sau khi đã gán các giá trị nhãn cho từng pixel.....	22
Hình 19: Hình kết hợp các nhãn lại với nhau	23
Hình 20: Hình kết quả cuối cùng có 2 phân vùng	23
Hình 21: Đường màu vàng trên hình là đường phân chia đối với tập dữ liệu gồm hai thuộc tính	25
Hình 22: Một bộ dữ liệu hai chiều được phân chia tuyến tính	26
Hình 23: 2 siêu phẳng phân chia tuyến tính cùng với biên độ của nó	27

Hình 24: Đường biểu diễn H_1 và H_2 , đường màu đỏ là khoảng cách Euclidean của hai điểm 1 và 2. Đường màu xanh là khoảng cách Euclidean nhỏ nhất	28
Hình 25: Các support vector trong SVM. Các support vector là những hình có viền dày hơn.	29
Hình 26: Một trường hợp đơn giản trên không gian 2 chiều	31
Hình 27: Mô hình bài toán	33

DANH MỤC BẢNG

Bảng 1: Bảng mô tả cơ sở dữ liệu	8
Bảng 2: Bảng đánh giá độ chính xác	35

CHƯƠNG 1 – MỞ ĐẦU

1.1 Giới thiệu đề tài

Nhận diện đối tượng trong thị giác máy tính là công việc tìm kiếm các vật thể trong ảnh. Đây là hướng nghiên cứu được quan tâm nhiều trong lĩnh vực công nghệ hiện nay với rất nhiều ứng dụng. Con người có thể nhận biết được các đối tượng từ ảnh một cách rất dễ dàng mặc dù sự khác nhau giữa chúng là vô cùng đa dạng. Tuy nhiên, việc này đối với hệ thống thị giác máy tính vẫn là một thách thức rất lớn. Nguyên nhân chủ yếu là do sự đa dạng về hình dáng, màu sắc của vật thể, đa dạng về diện mạo và tư thế của mỗi loại vật thể, ảnh hưởng về điều kiện ánh sáng, sự che lấp lẫn nhau giữa các vật thể, chất lượng ảnh, .v.v.

Ý tưởng chính trong các phương pháp nhận diện đối tượng là dựa vào đặc trưng cụ thể được rút trích ra từ những đối tượng mẫu. Đặc trưng này được sử dụng cùng với bộ phân lớp hoặc sử dụng thuật toán để nhận ra những đối tượng tương tự với các đối tượng mẫu.

Nhận diện biển số xe là một nhánh của nhận diện đối tượng.

1.2 Mục tiêu và phạm vi đề tài

Nhận diện biển số xe là bài toán có nhiều ứng dụng trong thực tế. Việc phát triển tốt, chính xác các biển số xe sẽ tạo tiền đề tốt cho những hướng phát triển như:

- Phục vụ quá trình quản lý xe tại bãi giữ xe.
- Dễ dàng tìm kiếm, tra cứu thông tin về xe, chủ xe tham gia giao thông, phục vụ cho các quá trình điều tra tội phạm, các trường hợp vi phạm giao thông.

Những khó khăn phải đối mặt trong đề tài này là:

- Các ảnh thường không đạt chuẩn, background phức tạp.
- Bất lợi về điều kiện ánh sáng.
- Thay đổi góc của vật thể trong ảnh.
- Sự che khuất của các vật khác lên đối tượng chính trong ảnh.

- Sự đa dạng về loại biển số, biển số xe ở các vùng, quốc gia khác nhau có thể khác nhau.

Dựa trên sự đa dạng của biển số xe ở các vùng, quốc gia khác nhau, phạm vi nhận dạng trong đề tài này được nhóm chia làm 2 loại: nhận dạng biển số có dạng hình chữ nhật và nhận dạng biển số có dạng hình vuông.

1.3 Cơ sở dữ liệu

Ký tự	Huấn luyện	Kiểm tra	Tổng
0	10	3	13
1	10	4	14
2	10	3	13
3	10	3	13
4	10	3	13
5	10	3	13
6	10	4	14
7	10	3	13
8	10	2	12
9	10	3	13
A	10	3	13
B	10	3	13
C	10	2	12
D	10	4	14
E	10	2	12

F	10	3	13
G	10	3	13
H	10	2	12
J	10	3	13
K	10	2	12
L	10	3	13
M	10	3	13
N	10	3	13
P	10	3	13
Q	10	3	13
R	10	2	12
S	10	3	13
T	10	5	15
U	10	3	13
V	10	3	13
W	10	4	14
X	10	3	13
Y	10	2	12
Z	10	3	13
Tổng	340	101	441

Bảng 1: Bảng mô tả cơ sở dữ liệu

CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

2.1 Khái niệm về ảnh

Quá trình rút trích đặc trưng là quá trình phân tích phổ (spectral analysis) nhằm xác định các thông tin quan trọng, đặc trưng của tín hiệu tiếng nói, tối thiểu hoá ảnh hưởng của nhiễu, xúc cảm, trạng thái của con người, cách phát âm của người nói, ...

Ảnh là một đối tượng sử dụng để lưu trữ bất kì thông tin về sự vật, hiện tượng ngoài đời sống mà con người có thể nhìn thấy. Để máy tính có thể xử lý được ảnh, cần phải đưa nó về dạng số hoá.

Ảnh được số hoá là ảnh được thể hiện trong không gian hai chiều mô tả bức ảnh gần với ảnh thật, với mỗi phần tử trong ma trận ảnh thể hiện cường độ sáng của bức ảnh tại điểm đó, được gọi là điểm ảnh.

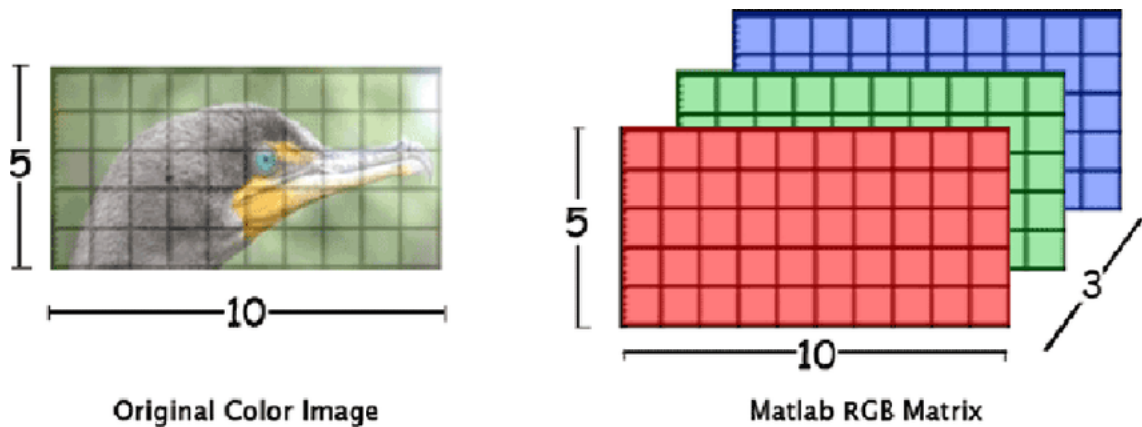
Mỗi điểm ảnh sẽ lưu giữ thông tin về mức xám (Gray level), được biểu thị bằng giá trị nguyên nằm trong khoảng 0 tới 255.

Một số định dạng ảnh bao gồm:

- Ảnh nhị phân: Mỗi điểm ảnh chỉ lưu trữ hai giá trị là 0 và 1.
- Ảnh đen trắng: Chỉ chứa 2 màu đen / trắng, ở mỗi điểm ảnh mức xám có thể khác nhau.
- Ảnh xám: Mỗi điểm ảnh lưu trữ một giá trị trong khoảng 0 tới 255.
- Ảnh màu: Gồm 3 ma trận ảnh chồng lên nhau, mỗi điểm ảnh lưu trữ 3 giá trị trong khoảng 0 tới 255.



Hình 1: Ví dụ về ảnh xám



Hình 2: Ví dụ ảnh màu trên không gian RGB (Red, Green, Blue)

2.2 Tiền xử lý

Do đặc điểm ảnh bao gồm số các điểm ảnh rất lớn, khiến cho việc tính toán nhiều, độ phức tạp cao, đòi hỏi dung lượng bộ nhớ lớn, thời gian tính toán lâu. Ngoài ra các ảnh khi đưa vào là các ảnh thô, có thể bị nhiễu, chi tiết thừa.

Chính vì thế, để có một bức ảnh tốt phục vụ cho quá trình nhận dạng được chính xác, cần phải trải qua công đoạn tiền xử lý. Mục tiêu của giai đoạn này là làm cho bức

ảnh trở nên rõ ràng hơn qua các bước khử nhiễu, tăng độ tương phản, đầu ra sẽ là bức ảnh đã được tinh chỉnh, phục vụ cho quá trình nhận dạng.

2.2.1 Chuyển đổi hệ màu

Chuyển hệ màu là bước chuyển từ bức ảnh màu sang các hệ màu khác. Tùy thuộc vào quá trình xử lý mà ta chuyển sang các hệ màu phù hợp. Trong đề tài này, nhóm sử dụng phương pháp chuyển ảnh màu thành ảnh xám trong bước tiền xử lý.

Một số phương pháp chuyển đổi ảnh màu sang ảnh xám:

- Lightness method: $I = \frac{(\max(R,G,B) + \min(R,G,B))}{2}$
- Average method: $I = \frac{R + G + B}{3}$
- Luminosity method: $I = (0.21 * R) + (0.72 * G) + (0.07 * B)$

Trong đó:

- I là giá trị mức xám tại 1 điểm ảnh (pixel)
- R là giá trị của pixel đó trên ma trận màu đỏ Red
- G là giá trị của pixel đó trên ma trận màu xanh Green
- B là giá trị của pixel đó trên ma trận màu xanh Blue



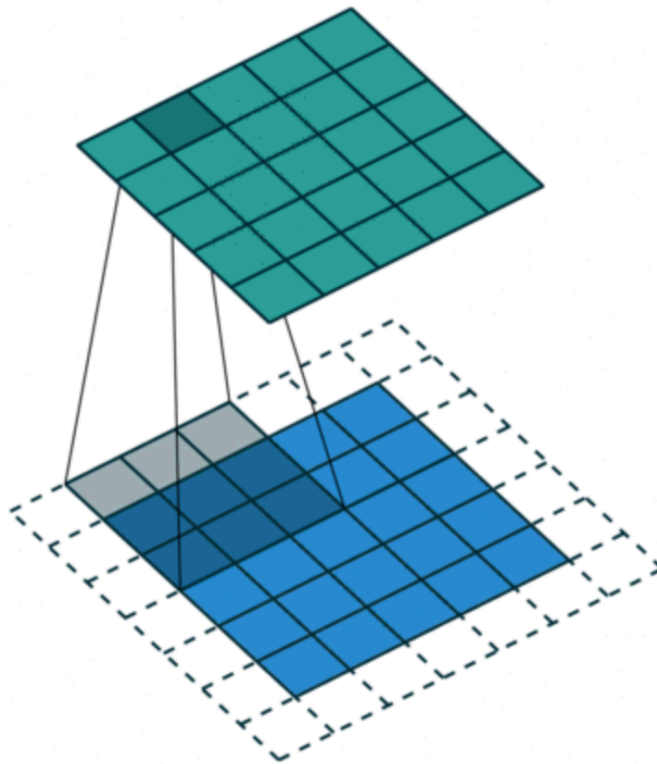
Hình 3: Ảnh đã chuyển qua ảnh xám

2.2.2 Lọc ảnh và khử nhiễu

Lọc ảnh (hay khử nhiễu) là quá trình làm mờ, mịn ảnh. Mục tiêu của bước này là loại bỏ các chi tiết nhiễu không cần thiết. Đối với một số tấm ảnh đã cũ, hoặc có những đốm màu trên ảnh, làm ảnh hưởng tới chi tiết của bức ảnh, ta cần loại bỏ chúng để có được bức ảnh trơn hơn, chỉ giữ lại các chi tiết chính trong ảnh.

Nguyên tắc chung của các phương pháp lọc là cho ma trận ảnh nhân với một ma trận lọc (Kernel). Ma trận lọc (Kernel) còn có thể được gọi là cửa sổ chập (trong phép nhân chập), cửa sổ lọc, mặt nạ,...

Việc nhân ảnh với ma trận lọc giống như việc trượt ma trận lọc theo hàng trên ảnh và nhân với từng vùng của ảnh, cộng các kết quả lại tạo thành kết quả của điểm ảnh trung tâm.



Hình 4: Minh họa việc nhân ma trận ảnh

Một số phương pháp lọc ảnh phổ biến:

- Bộ lọc trung bình (Normalized Box Filter)
- Bộ lọc Gaussian (Gaussian Filter)
- Bộ lọc trung vị (Median Filter)
- Bộ lọc hai chiều (Bilateral Filter) - Phương pháp chính được sử dụng trong đề tài này.

Bilateral Filter là bộ lọc làm mịn phi tuyến tính, bảo toàn cạnh và giảm nhiễu cho hình ảnh. Nó thay thế cường độ của từng pixel bằng giá trị cường độ trung bình có trọng số từ các pixel lân cận.

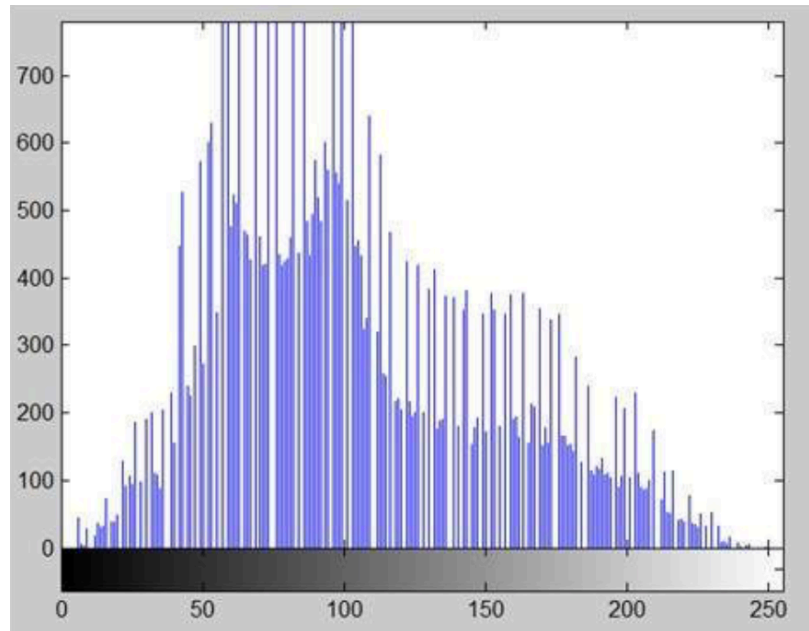
Bộ lọc Bilateral cũng sử dụng một bộ lọc Gauss với khoảng cách đến điểm trung tâm, đảm bảo chỉ có các điểm ở gần tham gia vào giá trị của điểm ảnh trung tâm. Tuy vậy nó sử dụng thêm một hàm Gauss cho mức xám, đảm bảo chỉ các điểm ảnh có mức xám tương đồng với điểm ảnh trung tâm tham gia vào quá trình làm mịn. Vì thế bộ lọc Bilateral bảo toàn được các đường biên trong ảnh bởi vì điểm ảnh ở biên có sự thay đổi về mức xám rất rõ ràng.



Hình 5: Kết quả sau khi sử dụng bộ lọc Bilateral

2.2.3 Cân bằng sáng (*Histogram Equalization*)

Histogram của ảnh là biểu đồ cột thống kê số lần xuất hiện của các mức sáng trong ảnh. Trong biểu đồ này, trục hoành sẽ biểu thị giá trị của mức xám $[0,255]$, trục tung sẽ là tần suất xuất hiện của các mức xám đó trong ảnh.



Hình 6: Ví dụ về histogram

Phương pháp cân bằng histogram thường làm tăng độ tương phản toàn cầu của nhiều hình ảnh, đặc biệt là khi dữ liệu có thể sử dụng của hình ảnh được biểu thị bằng các giá trị tương phản gần. Thông qua điều chỉnh này, cường độ có thể được phân phối tốt hơn trên biểu đồ. Điều này cho phép các khu vực có độ tương phản cục bộ thấp hơn để đạt được độ tương phản cao hơn. Cân bằng biểu đồ thực hiện điều này bằng cách trải rộng hiệu quả các giá trị cường độ thường xuyên nhất.

Một số công dụng của việc cân bằng histogram:

- Giảm sự ảnh hưởng do ảnh quá sáng hoặc quá tối.
- Giúp “chuẩn hoá” ảnh đầu vào trước khi tiến hành xử lý.



Hình 7: Kết quả sau khi thực hiện cân bằng sáng Histogram Equalization

2.2.4 Xử lý hình thái - *Morphological*

Xử lý hình ảnh hình thái là một tập hợp các hoạt động phi tuyến tính liên quan đến hình dạng hoặc hình thái của các đặc trưng trong một hình ảnh. Các hoạt động hình thái chỉ dựa vào thứ tự tương đối của các giá trị pixel, chứ không dựa trên các giá trị số của chúng và do đó đặc biệt phù hợp với việc xử lý ảnh nhị phân.

Xử lý hình thái sử dụng một dạng hình khối để dò hình ảnh được gọi là phần tử cấu trúc. Phần tử cấu trúc được định vị tại tất cả các vị trí có thể có trong hình ảnh và nó được so sánh với vùng lân cận pixel tương ứng. Một số thao tác kiểm tra xem phần tử có "khớp" trong vùng lân cận hay không, trong khi các phần tử khác kiểm tra xem phần tử đó có "chạm" hay giao với vùng lân cận hay không.

Phần tử cấu trúc là một ma trận nhị phân nhỏ, mỗi phần tử trong ma trận là một pixel có giá trị bằng 0 hoặc 1, trong đó:

- Kích thước ma trận xác định kích thước của phần tử cấu trúc.
- Góc của phần tử cấu trúc thường là một trong các pixel của nó, mặc dù nhìn chung góc có thể nằm ngoài phần tử cấu trúc.

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Hình 8: Phần tử cấu trúc có kích thước 5x5

Trên thực tế các phần tử cấu trúc có kích thước lẻ (3x3, 5x5, ...) và gốc được xác định là tâm của ma trận. Phần tử cấu trúc hoạt động tương tự như ma trận tích chập (Kernel) trong các phương pháp lọc ảnh.

Khi một phần tử cấu trúc được đặt trong một hình ảnh nhị phân, mỗi pixel của nó được liên kết với pixel tương ứng của vùng lân cận dưới phần tử cấu trúc.

Phần tử cấu trúc được gọi là “fit” với hình ảnh nếu ứng với mỗi pixel của nó được gán bằng 1, pixel ở hình ảnh tương ứng cũng bằng 1. Tương tự, một phần tử cấu trúc được cho là “hit” hoặc “intersect” nếu ít nhất 1 trong các pixel của phần tử cấu trúc được gán bằng 1 thì pixel của ảnh tương ứng cũng bằng 1.

00000000000000									
B	00	11	00	00	00	00	00	00	00
	00	11	11	11	00	00	00	00	00
	01	11	11	11	10	00	00	00	00
	01	11	11	11	10	00	00	00	00
	00	11	11	11	10	00	00	00	00
	00	11	11	11	11	10	00	00	00
	00	11	11	11	11	11	10	00	00
A	00	00	00	11	11	11	10	00	00
	00	00	00	00	00	00	00	00	00

$s_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$
 $s_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$

		A	B	C
fit	s ₁	yes	no	no
	s ₂	yes	yes	no
hit	s ₁	yes	yes	yes
	s ₂	yes	yes	no

Hình 9: “fit” và “hit” của ảnh nhị phân với phần tử cấu trúc s1 và s2

2.2.4.1 Phương pháp ăn mòn (Erosion)

Ý tưởng cơ bản về ăn mòn cũng giống như ăn mòn đất, nó ăn mòn biên của đối tượng tiền cảnh (Luôn cố gắng giữ tiền cảnh trong màu trắng).

Phần tử cấu trúc sẽ trượt qua hình ảnh, như trong phương pháp tích chập, một pixel trong ảnh gốc (0 hoặc 1) sẽ chỉ được có là 1 nếu tất cả các pixel trong phần tử cấu trúc là 1, nếu không nó sẽ bị ăn mòn (được gán là 0).

Vì vậy, tất cả các pixel gần biên sẽ bị loại bỏ tùy thuộc vào kích thước của phần tử cấu trúc. Độ dày hoặc kích thước của đối tượng chính sẽ bị giảm, nói một cách đơn giản, những vùng màu trắng trong tấm hình sẽ bị giảm bớt.

Phương pháp này rất tốt cho việc loại bỏ những hạt nhiễu màu trắng, tách 2 đối tượng kết nối nhau, v.v...



Hình 10: Ảnh gốc (bên trái) và ảnh sau khi sử dụng phương pháp ăn mòn (bên phải).

2.2.4.2 Phương pháp giãn nở (Dilation)

Phương pháp này chỉ đơn giản là ngược với phương pháp ăn mòn ở trên. Ở đây, một phần tử pixel là 1 nếu ít nhất một pixel trong phần tử cấu trúc là 1. Vì vậy, nó làm tăng vùng trắng trong hình ảnh hoặc làm kích thước của đối tượng chính trong ảnh tăng lên.

Thông thường, trong các trường hợp khử nhiễu, giãn nở thường đi kèm với ăn mòn trước đó. Bởi vì ăn mòn có thể loại bỏ hạt nhiễu trắng, nhưng nó cũng ăn bớt chủ

thể chính. Vì vậy, phương pháp giãn nở được sử dụng ngay sau đó, nó giúp cho kích thước đối tượng chủ thể tăng trở lại.

Phương pháp này rất có ích trong việc nối các phần bị mất của đối tượng.



Hình 11: Ảnh gốc (bên trái) và ảnh sau khi sử dụng phương pháp giãn nở (bên phải).

2.2.4.3 Phương pháp mở (Opening)

Đây là phương pháp mở rộng, nó kết hợp 2 phương pháp ăn mòn và giãn nở. Cụ thể nó sẽ thực hiện ăn mòn, sau đó sẽ giãn nở. Mục đích để khử những hạt nhiễu không cần thiết.



Hình 12: Kết quả sau khi thực hiện phương pháp Opening.

2.2.4.4 Phương pháp đóng (Closing)

Phương pháp này sẽ ngược với phương pháp Opening, thực hiện giãn nở trước, tiếp sau đó là ăn mòn. Ứng dụng của phương pháp này thường dùng để phục hồi các phần bị mất của đối tượng.



Hình 13: Kết quả sau khi thực hiện phương pháp Closing.

2.2.5 Phương pháp lấy ngưỡng

Ngưỡng dùng để chỉ một giá trị được sử dụng để phân hoạch một tập hợp thành các miền phân biệt. Trong xử lý ảnh, ngưỡng được sử dụng để phân hoạch các giá trị của pixel. Nếu như pixel có giá trị lớn hơn ngưỡng, thì nó được gán giá trị là 1, ngược lại là 0.

Đầu ra của phương pháp này sẽ là một ảnh dạng nhị phân. Phương pháp này giúp ta loại bỏ bớt những chi tiết không cần thiết.



Hình 14: Ảnh sau khi lấy ngưỡng

2.3 Connected Component Analysis (CCA)

Phương pháp tìm vùng liên kết hoạt động bằng cách quét toàn bộ bức ảnh từ trên xuống dưới, trái qua phải, sau đó nhóm các pixel liên kề có sự liên kết với nhau. Các pixel được gọi là liên kết khi chúng có cường độ tương tự nhau.

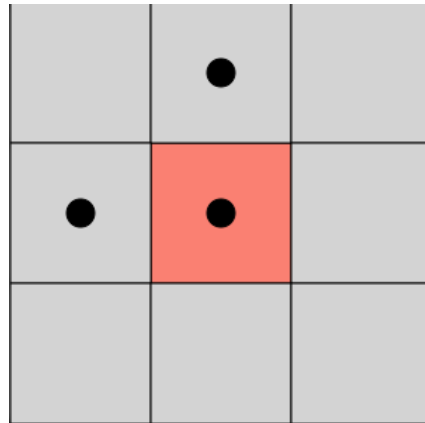
Thuật toán này thường được áp dụng trên các bức ảnh nhị phân hoặc ảnh xám, khi đó các pixel chỉ có 2 giá trị là 0 và 1. Giải thuật sẽ quét bức ảnh và lấy ra những vùng nào có cùng giá trị là 0 hoặc 1. Trong ảnh xám sẽ sắp xếp các tập giá trị.

Ví dụ như $V = \{51, 52, 53, \dots, 77, 78, 79, 80\}$

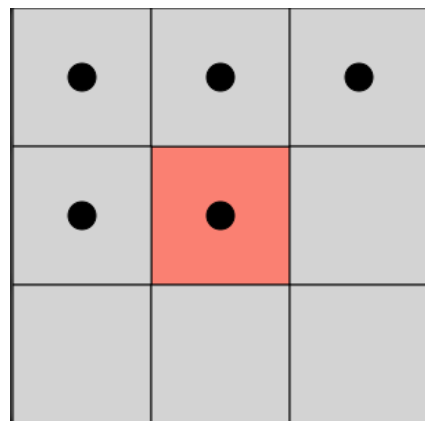
Ví dụ như hình ảnh đầu vào nhị phân và 8-connectivity. Toán tử ghi nhãn các thành phần được kết nối sẽ quét hình ảnh bằng cách di chuyển dọc theo một hàng cho đến khi đến điểm p (trong đó p biểu thị pixel được gắn nhãn ở bất kỳ giai đoạn nào trong quy trình quét) mà $V = \{1\}$. Khi điều này là đúng, nó sẽ kiểm tra bốn lân cận của p đã gặp trong quá trình quét (tức là các lân cận (i) ở bên trái của p , (ii) phía trên nó và (iii và iv) hai đường chéo trên). Dựa trên thông tin này, việc ghi nhãn p xảy ra như sau:

- Nếu 4 lân cận đều là 0 thì gán giá trị nhãn mới cho p.
- Nếu chỉ có 1 lân cận mà $V = \{1\}$ thì gán nhãn của lân cận đó cho p.
- Nếu nhiều hơn 1 lân cận có $V = \{1\}$ thì gán một nhãn cho p và ghi chú lại chúng bằng nhau.

Sau khi hoàn thành quá trình quét, các cặp nhãn tương đương được sắp xếp thành các lớp tương đương và một nhãn duy nhất được gán cho mỗi lớp. Bước cuối cùng, lần quét thứ hai được thực hiện thông qua hình ảnh, trong đó mỗi nhãn được thay thế bằng nhãn được gán cho các lớp tương đương của nó. Để hiển thị, các nhãn có thể là màu xám hoặc màu khác nhau.



Hình 15: Lận cận 4



Hình 16: Lận cận 8

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1	1	0	0	1	1	1	1	0	0
0	0	0	1	1	1	1	0	0	0	1	1	1	1	0	0	0
0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	0
0	1	1	1	0	0	1	1	0	0	0	1	1	1	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	1	1	0
0	0	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

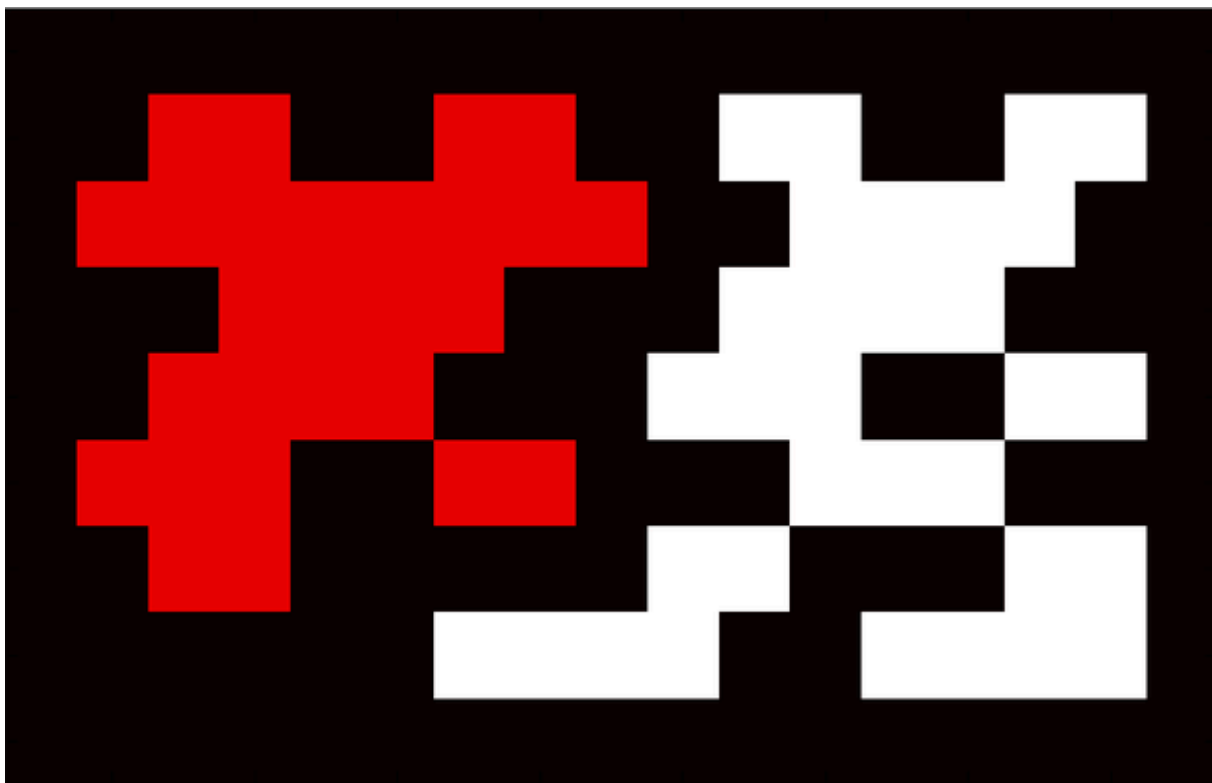
Hình 17: Hình ảnh đầu vào

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	2	2	0	0	3	3	0	0	4	4	0
0	1	1	1	1	1	1	1	1	0	0	3	3	3	3	0	0
0	0	0	1	1	1	1	0	0	0	3	3	3	3	0	0	0
0	0	1	1	1	1	0	0	0	3	3	3	0	0	3	3	0
0	1	1	1	0	0	1	1	0	0	0	3	3	3	0	0	0
0	0	1	1	0	0	0	0	0	5	3	0	0	0	3	3	0
0	0	0	0	0	0	6	6	5	3	0	0	7	3	3	3	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Hình 18: Hình sau khi đã gán các giá trị nhãn cho từng pixel

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	1	0	0	3	3	0	0	3	3	0
0	1	1	1	1	1	1	1	1	0	0	3	3	3	3	0	0
0	0	0	1	1	1	1	0	0	0	3	3	3	3	0	0	0
0	0	1	1	1	1	0	0	0	3	3	3	0	0	3	3	0
0	1	1	1	0	0	1	1	0	0	0	3	3	3	0	0	0
0	0	1	1	0	0	0	0	0	3	3	0	0	0	3	3	0
0	0	0	0	0	0	3	3	3	3	0	0	3	3	3	3	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Hình 19: Hình kết hợp các nhãn lại với nhau



Hình 20: Hình kết quả cuối cùng có 2 phân vùng

2.4 Support Vector Machine

2.4.1 Giới thiệu:

SVM là một phương pháp trong việc phân loại dữ liệu tuyến tính và không tuyến tính. Có nhiều phương pháp phân loại dữ liệu như: phân lớp dựa vào cây quyết định, phân lớp dựa vào luật, phân lớp Bayesian, phân lớp theo lan truyền ngược... Tuy nhiên trong lĩnh vực của đề tài thì SVM là phương pháp phân lớp được chọn để sử dụng.

Bài báo đầu tiên về Support Vector Machine được giới thiệu vào năm 1992 bởi Vladimir Vapnik và hai đồng sự Bernhard Boser và Isabelle Guyon, mặc dù nền móng cơ bản của SVM đã có từ năm 1960 (bao gồm các công việc được thực hiện rất sớm bởi Vapnik và Alexei Chervonenkis trong lý thuyết học thống kê).

Trước khi đi vào tìm hiểu phương pháp SVM, ta phải biết được các khái niệm về siêu phẳng phân chia tuyến tính, support vector...

2.4.2 Các khái niệm cơ bản:

2.4.2.1 Siêu phẳng phân cách:

Cho trước tập dữ liệu D gồm $(X_1, y_1), (X_2, y_2), \dots, (X_{|D|}, y_{|D|})$.

Trong đó X_i là một tập các bộ huấn luyện tương ứng với nhãn lớp y_i . Mỗi y_i sẽ nhận một trong hai giá trị hoặc là $+1$ hoặc là -1 ($y_i \in \{+1, -1\}$).

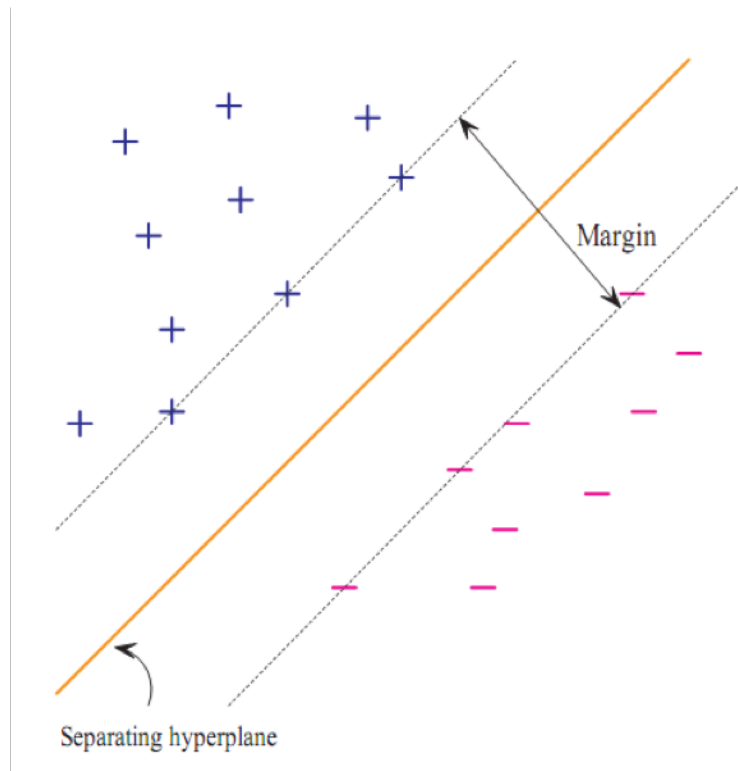
Phương pháp phân lớp SVM sẽ tìm ra đường phân lớp “tốt nhất” để phân chia tập dữ liệu này thành từng lớp tách biệt ra với nhau. Theo tài liệu “Robust Real-time Object Detection” của Paul Viola và Michael Jones, phương trình tổng quát của một đường phân chia như vậy được biểu diễn dưới dạng sau:

$$W \cdot X + b = 0 \quad (1)$$

Trong đó:

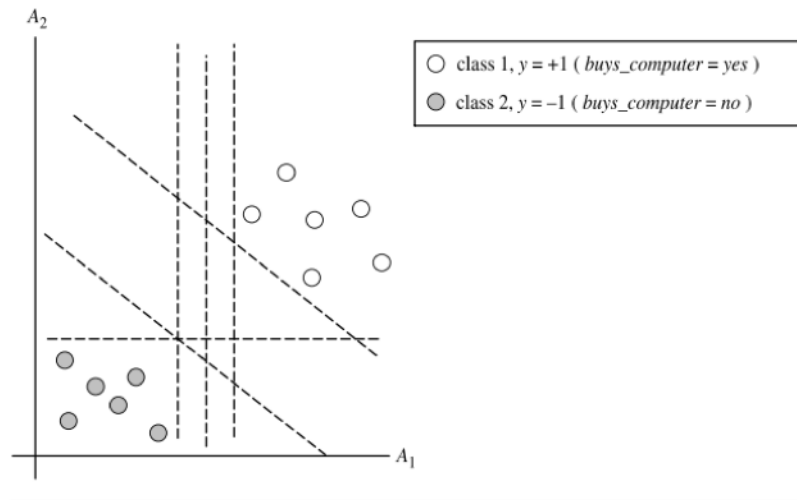
- W : Vector trọng số, $W = \{w_1, w_2, \dots, w_n\}$.
- n : Số thuộc tính (hay còn gọi là số chiều của dữ liệu).
- b : Một đại lượng vô hướng, thường được xem như là một độ nghiêng (bias).

Đối với trường hợp dữ liệu hai chiều (hai thuộc tính) thì phương trình trên biểu diễn của đường thẳng phân chia. Nếu dữ liệu của chúng ta là ba chiều thì đường phân chia giữa hai tập sẽ là một mặt phẳng phân cách. Tổng quát cho dữ liệu n chiều thì sẽ được phân cách bởi một siêu phẳng. Chúng ta sẽ sử dụng thuật ngữ “siêu phẳng” (hyperplane) để chỉ đến ranh giới quyết định mà chúng ta muốn tìm kiếm bất chấp số lượng thuộc tính.



Hình 21: Đường màu vàng trên hình là đường phân chia đối với tập dữ liệu gồm hai thuộc tính

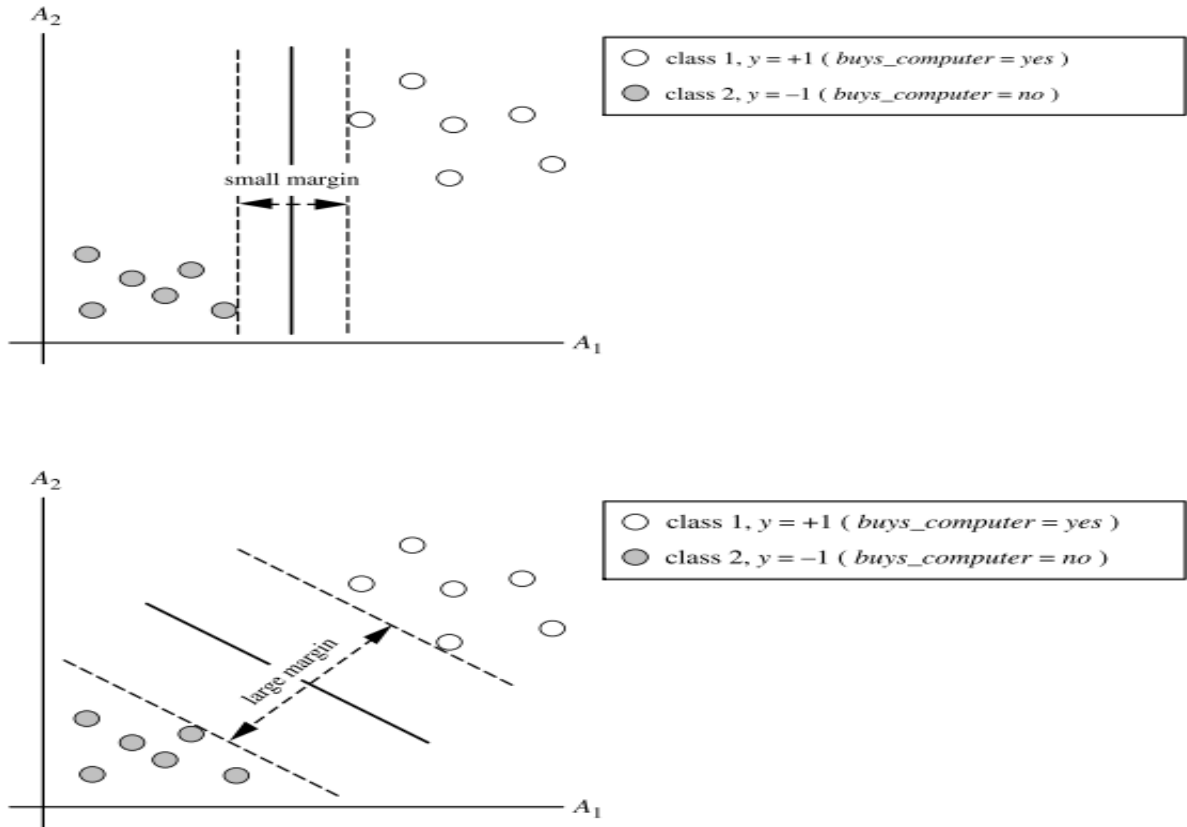
Tuy nhiên trong thực tế ta có thể tìm được vô số những siêu phẳng phân chia trên cùng một tập dữ liệu. Và chúng ta muốn tìm đường thẳng phân chia sao cho tốt nhất, có nghĩa là có sai sót phân loại bé nhất trên bộ dữ liệu.



Hình 22: Một bộ dữ liệu hai chiều được phân chia tuyến tính

Do đó mục tiêu của phương pháp phân lớp SVM là tìm một siêu phẳng phân cách giữa hai lớp sao cho khoảng cách lề (margin) giữa hai lớp đạt cực đại.

Siêu phẳng có biên độ lớn nhất (maximum marginal hyperplane) sẽ được chọn như là siêu phẳng phân chia tập dữ liệu một cách tốt nhất. Trong hình bên dưới, ta thấy có hai siêu phẳng có thể phân chia được và những biên độ của nó. Trước khi đi vào định nghĩa của biên độ (margin), hãy nhìn vào hình trên một cách trực quan. Cả hai siêu phẳng đều phân tách tất cả những bộ dữ liệu cho trước. Một cách trực quan, siêu phẳng với biên độ lớn hơn sẽ chính xác hơn trong việc phân loại các bộ dữ liệu trong tương lai so với siêu phẳng có biên độ nhỏ hơn. Điều này là lý do tại sao (trong suốt giai đoạn học hay huấn luyện), SVM tìm những siêu phẳng có biên độ lớn nhất, gọi là MMH (maximum marginal hyperlane). Siêu phẳng có biên độ lớn nhất là siêu phẳng có khoảng cách từ nó tới hai mặt bên của nó thì bằng nhau (mặt bên song song với siêu phẳng). Khoảng cách đó thật ra là khoảng cách ngắn nhất từ MMH tới bộ dữ liệu huấn luyện gần nhất của mỗi lớp. Siêu phẳng có biên độ lớn nhất này cho chúng ta một sự phân loại tốt nhất giữa các lớp.



Hình 23: 2 siêu phẳng phân chia tuyến tính cùng với biên độ của nó

Siêu phẳng phân cách có vai trò quan trọng trong việc phân lớp, nó quyết định xem một bộ dữ liệu sẽ thuộc về lớp nào. Để thực hiện việc phân lớp, SVM chỉ cần xác định xem một bộ dữ liệu nằm về phía nào của siêu phẳng phân cách:

$$D(x) = \text{sign}(W \cdot X + b) \quad (2)$$

- $D(x) < 0$: bộ dữ liệu sẽ nằm phía dưới siêu phẳng phân cách
- $D(x) = 0$: bộ dữ liệu sẽ nằm trên siêu phẳng phân cách
- $D(x) > 0$: bộ dữ liệu sẽ nằm phía trên siêu phẳng phân cách

2.4.2.2 Support Vector:

Ta có phương trình tổng quát của siêu phẳng:

$$W \cdot X + b = 0$$

Ta xét trên ví dụ sau:

Với bộ dữ liệu huấn luyện có hai thuộc tính A_1 và A_2 : $X = \{x_1, x_2\}$, với x_1, x_2 là giá trị của thuộc tính A_1, A_2 . $W = \{w_1, w_2\}$. Phương trình siêu phẳng có thể viết lại:

$$w_0 + w_1x_1 + w_2x_2 = 0$$

Trong đó:

- w_0 tương đương với hằng số b trong phương trình tổng quát của siêu phẳng

Vì vậy mỗi điểm nằm trên siêu phẳng phân cách thỏa mãn:

$$w_0 + w_1x_1 + w_2x_2 > 0$$

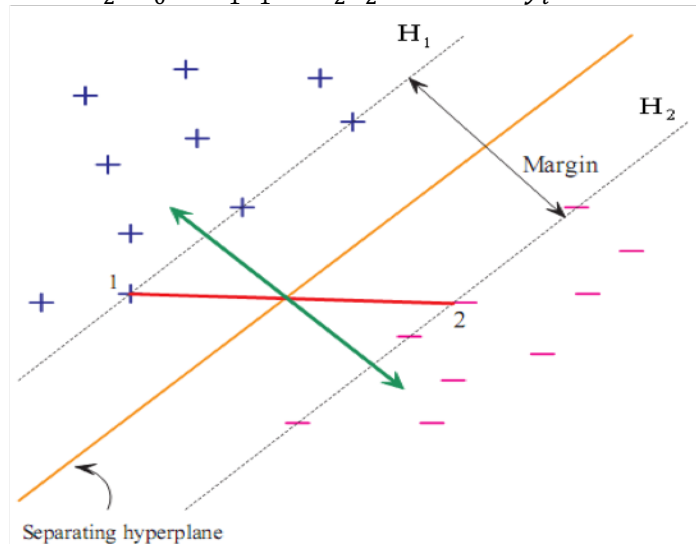
Tương tự, những điểm nằm dưới siêu phẳng phân cách phải thỏa mãn:

$$w_0 + w_1x_1 + w_2x_2 < 0$$

Bằng cách điều chỉnh trọng số w_0 ta có:

$$H_1: w_0 + w_1x_1 + w_2x_2 \geq 1 \text{ với } y_i = +1$$

$$H_2: w_0 + w_1x_1 + w_2x_2 \leq -1 \text{ với } y_i = -1$$



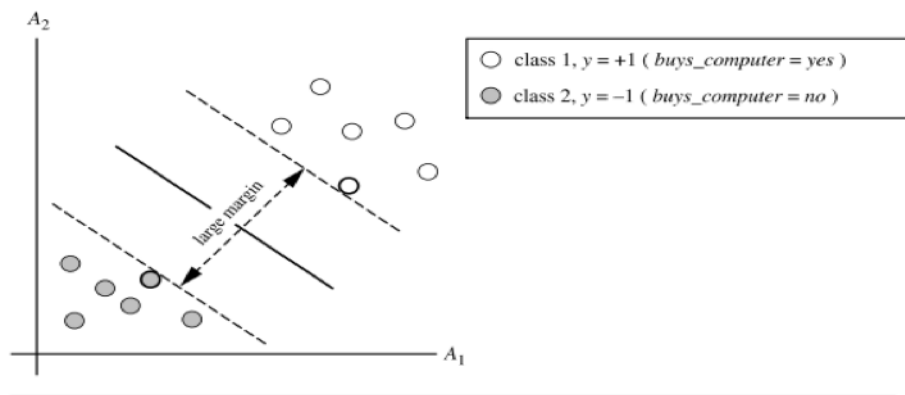
Hình 24: Đường biểu diễn H_1 và H_2 , đường màu đỏ là khoảng cách Euclidean của hai điểm 1 và 2. Đường màu xanh là khoảng cách Euclidean nhỏ nhất

Điều này có nghĩa là nếu bất kì bộ nào nằm tại hoặc trên H_1 đều thuộc về lớp +1, và bất kì bộ nào nằm tại hoặc dưới H_2 đều thuộc về lớp -1. Kết hợp 2 bất đẳng thức trên ta có:

$$y_i(w_0 + w_1x_1 + w_2x_2) \geq 1, \forall i$$

Mỗi bộ huấn luyện nằm tại các mặt biên H_1 hay H_2 thỏa mãn phương trình trên được gọi là support vectors. Support vectors là những bộ gần với siêu phẳng phân chia tuyến tính (MMH) nhất.

Trong hình bên dưới, support vectors là hình tròn có viền dày hơn. Ta thấy rằng các support vectors là những bộ khó phân lớp nhất và cung cấp nhiều thông tin nhất cho việc phân lớp.



Hình 25: Các support vector trong SVM. Các support vector là những hình có viền dày hơn.

2.4.2.3 Biên độ Margin:

Từ các điều trên chúng ta có thể đưa ra một công thức cho việc tính biên độ lớn nhất. Khoảng cách từ siêu phẳng phân chia đến mọi điểm tại H_1 là $\frac{1}{||W||}$

Trong đó:

- $||W||$ là khoảng cách Euclidean chuẩn của W là $\sqrt{W * W}$. Với $W = \{w_1, w_2\}$ khi đó $\sqrt{W * W} = \sqrt{w_1^2 + w_2^2}$.

Theo định nghĩa, khoảng cách từ siêu phẳng đến H_1 bằng với khoảng cách từ mọi điểm tại H_2 đến siêu phẳng. Vì vậy, kích thước của biên độ cực đại là $\frac{2}{\|w\|}$.

2.4.3 Phân lớp dữ liệu:

Trường hợp dữ liệu có thể phân chia tuyến tính được:

Việc huấn luyện SVM với mục đích là để tìm ra các support vectors và MMH. MMH là ranh giới phân chia tuyến tính giữa các lớp và vì thế SVM tương ứng có thể được sử dụng để phân lớp dữ liệu mà dữ liệu đó có thể phân chia tuyến tính. Chúng ta xem SVM được huấn luyện là SVM tuyến tính.

Sau khi huấn luyện SVM, chúng ta sẽ phân loại các bộ mới. Dựa trên công thức Lagrangian ta có:

$$d(X^T) = \sum_{i=1}^l y_i \alpha_i X^T + b_0$$

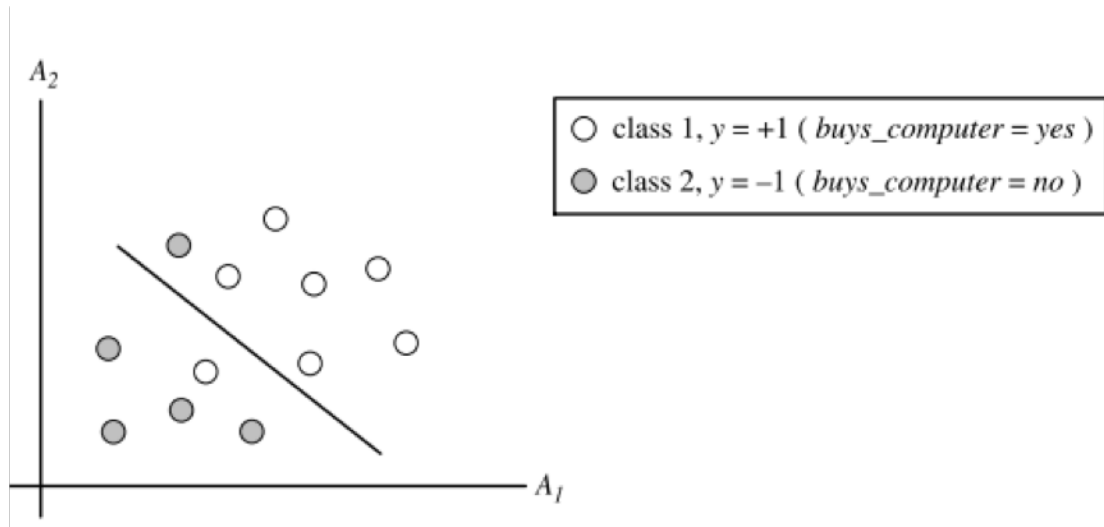
Trong đó:

- y_i là nhãn lớp của support vector X_i
- X^T là một bộ test
- α (nhân tử Lagrangian)
- b_0 là biến số được xác định bởi sự tối ưu hóa hay các thuật toán SVM
- l là số lượng các support vectors.

MMH có thể được xem như “ranh giới quyết định” trong việc quyết định xem một bộ test bất kỳ sẽ thuộc vào lớp nào. Cho một bộ test X^T , chúng ta gán nó vào phương trình trên, và sau đó kiểm tra dấu của kết quả. Từ đó ta sẽ biết được bộ test sẽ rơi vào mặt nào của siêu phẳng. Nếu dấu là dương, thì X^T rơi vào phía trên của MMH, và SVM đoán rằng X^T thuộc về lớp +1. Nếu dấu là âm, thì X^T nằm tại hoặc dưới MMH và nhãn lớp được đoán là -1.

2.4.3.2 Trường hợp dữ liệu không thể chia tuyến tính được:

Trong phần trên chúng ta đề cập đến trường hợp SVM phân lớp những dữ liệu có thể phân chia tuyến tính, nhưng nếu dữ liệu không thể phân chia tuyến tính thì sao? Trong trường hợp này không có đường thẳng nào có thể vẽ được để phân chia các lớp này. SVM tuyến tính mà chúng ta đã học thì không đem lại lời giải khả thi trong trường hợp này.



Hình 26: Một trường hợp đơn giản trên không gian 2 chiều

Tuy nhiên hướng tiếp cận của SVM tuyến tính có thể được mở rộng để tạo ra SVM không tuyến tính cho việc phân lớp các dữ liệu không thể phân chia tuyến tính (hay gọi tắt là dữ liệu không tuyến tính). Những SVM như vậy có khả năng tìm những ranh giới quyết định không tuyến tính (những mặt không tuyến tính) trong không gian đầu vào.

“Làm thế nào mở rộng tiếp cận tuyến tính?”. Chúng ta thu được SVM phi tuyến bằng cách mở rộng SVM tuyến tính như sau. Có hai bước chính:

- Bước 1: Ta chuyển dữ liệu nguồn lên một không gian nhiều chiều hơn bằng cách sử dụng ánh xạ phi tuyến. Một vài ánh xạ phi tuyến thông thường có thể được sử dụng để thực hiện bước này

- Bước 2: Tìm những siêu phẳng trong không gian mới này. Cuối cùng chúng ta lại quay lại vấn đề tối ưu bình phương đã được giải quyết sử dụng công thức SVM tuyến tính. Siêu phẳng có biên độ lớn nhất được tìm thấy trong không gian mới tương ứng với siêu bề mặt phân chia không tuyến tính trong không gian ban đầu.

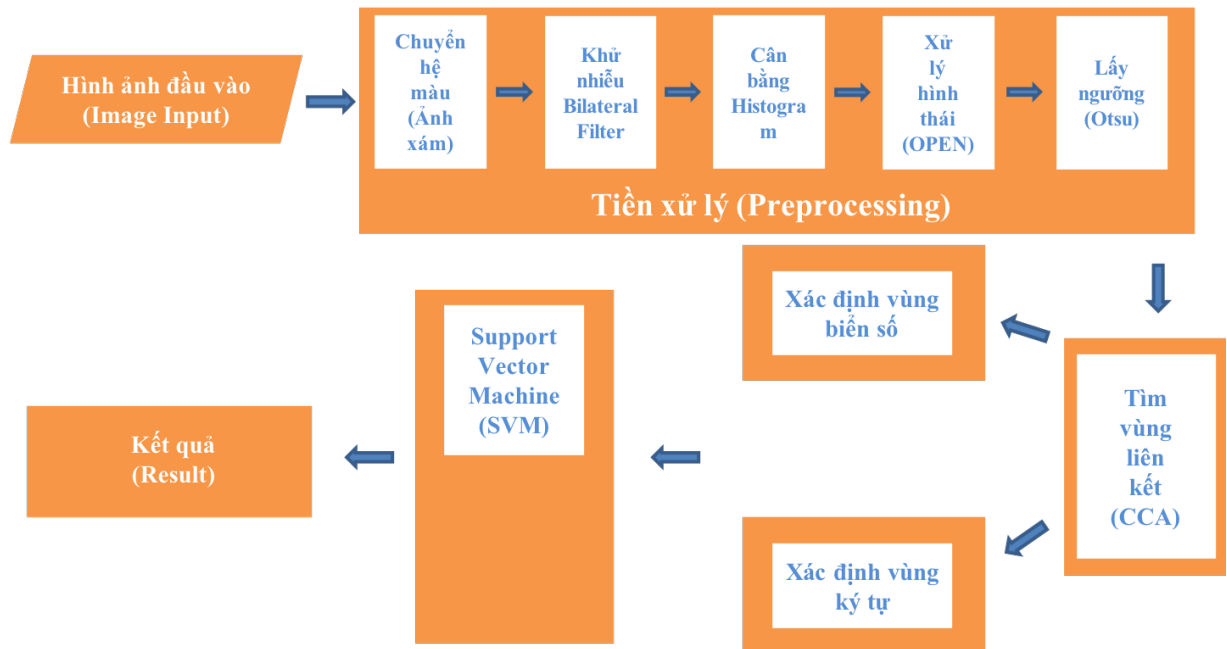
CHƯƠNG 3 – MÔ HÌNH GIẢI QUYẾT BÀI TOÁN

3.1 Giới thiệu

Hệ thống nhận dạng biển số xe được xây dựng thông qua 2 công đoạn chính, cụ thể là:

- Giai đoạn tiền xử lý và rút trích thông tin cần nhận dạng.
- Giai đoạn huấn luyện dữ liệu.

3.2 Chi tiết mô hình



Hình 27: Mô hình bài toán

CHƯƠNG 4 – KẾT QUẢ THỰC NGHIỆM

4.1 Phương pháp đánh giá

Bài toán được đánh giá dựa trên công thức sau đây:

$$\text{Độ chính xác (\%)} = \frac{\text{Số lượng ảnh đúng} \times 100\%}{\text{Tổng số lượng ảnh}}$$

Phương pháp chạy thuật toán là kiểm tra chéo (cross validation): chạy 5 lần với mỗi lần tập huấn luyện (70% số lượng ảnh) và tập ảnh kiểm tra (30% số lượng ảnh) là khác nhau để tránh trường hợp quá khớp dữ liệu (over-fitting).

- Bộ 1: tập huấn luyện (1,2,3,4,5,6,7), tập kiểm tra (8,9,10).
- Bộ 2: tập huấn luyện (4,5,6,7,8,9,10), tập kiểm tra (1,2,3).
- Bộ 3: tập huấn luyện (1,2,3,4,5,7,8), tập kiểm tra (4,5,6).
- Bộ 4: tập huấn luyện (2,3,4,5,6,8,10), tập kiểm tra (1,7,9).
- Bộ 5: tập huấn luyện (2,3,5,6,7,9,10), tập kiểm tra (1,4,8).

4.2 Kết quả thực nghiệm

Tổng số tập huấn luyện: 340

Tổng số tập kiểm tra: 130

Số lần	Tỉ lệ đúng
1	85,65%
2	71,85%
3	85%

4	87%
5	85%

Bảng 2: Bảng đánh giá độ chính xác

CHƯƠNG 5 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Các hạn chế

Dựa trên những đánh giá từ kết quả thu được, có thể thấy những mặt hạn chế của ứng dụng như sau:

- Bộ cơ sở dữ liệu về biển số xe còn ít, ứng dụng chỉ mới nhận dạng được một số ít biển số xe.
- Ứng dụng không nhận dạng các biển số xe quá xa, quá gần hoặc bị thay đổi góc bức ảnh.

5.2 Những điều đã đạt được

Tuy ứng dụng còn nhiều mặt hạn chế, nhưng thông qua quá trình nghiên cứu và áp dụng, nhóm chúng em cơ bản đã thực hiện được một số điểm:

- Tìm hiểu về các phương pháp tiền xử lý ảnh, rút đặc trưng và huấn luyện mô hình.
- Xây dựng bộ cơ sở dữ liệu nhỏ về biển số xe.
- Xây dựng thành công chương trình nhận dạng biển số xe trên hình ảnh hoặc video.

5.3 Hướng phát triển

Những kết quả thu được chính là nền tảng quý báu, từ đó có thể phát triển thêm một số hướng cho đề án như:

- Mở rộng bộ cơ sở dữ liệu, hướng đến mục tiêu nhận dạng thêm nhiều biển số, với đa dạng góc ảnh để hệ thống đạt độ chính xác cao hơn.
- Tìm hiểu và áp dụng thêm các kỹ thuật học máy thích hợp vào quá trình xây dựng mô hình, tiến hành thực nghiệm để tìm ra mô hình tối ưu với độ chính xác cao.

TÀI LIỆU THAM KHẢO

Tiếng Việt

1. Lý thuyết về phân ngưỡng ảnh (threshold) - <https://thigiacmaytinhtinh.com/ly-thuyet-ve-phan-nguong-anh-threshold/>

Tiếng Anh

1. GitHub LicensePlateDetector - https://github.com/apoorva-dave/LicensePlateDetector?fbclid=IwAR37YdZwsZLMo_DTwtbHglIwCRKpqxBwy4awiIW4Fm8KMqp3B0turMYg0M
2. Morphological Transformations - https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html
3. Morphological Image Processing - <https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm>
4. Connected-component labeling - https://en.wikipedia.org/wiki/Connected-component_labeling