

LỜI CẢM ƠN

Trong quá trình học tập tại trường, chúng tôi đã được các giảng viên truyền đạt những kiến thức vô cùng quý báu với sự tận tâm, tận tình và sự tận tụy với nghề, với sinh viên của quý thầy cô. Chúng tôi được trao đổi kiến thức qua từng môn học cũng như trải nghiệm, học hỏi và cung cấp kiến thức qua từng đề tài đồ án môn học. Với sự động viên, giúp đỡ tận tình từ các thầy cô, gia đình và bạn bè, đã giúp chúng tôi tiến bước đến ngày hôm nay và hoàn thành đồ án tốt nghiệp một cách trọn vẹn.

Với sự hoàn thành của đề tài “Nghiên cứu, thiết kế và chế tạo máy và xử lý nhận dạng nhân trắc (tai-mũi)”. Nhóm chúng tôi xin gửi lời tri ân sâu sắc đến quý thầy cô trường Đại học Sư phạm Kỹ thuật TP. Hồ Chí Minh nói chung và các thầy cô khoa Cơ khí Chế tạo máy nói riêng đã tận tình hướng dẫn, truyền đạt những kiến thức, kinh nghiệm quý báu từ lúc chúng tôi mới bắt đầu quá trình thực hiện cho đến khi hoàn thành đề tài. Và sự hỗ trợ về kiến thức chuyên môn cùng nguồn dữ liệu từ Dr.Tuấn

Đặc biệt, chúng tôi xin gửi lời cảm ơn chân thành đến giảng viên hướng PGS.TS Nguyễn Trường Thịnh, là người trực tiếp chỉ bảo, giúp đỡ chúng tôi trong suốt quá trình thực hiện đề tài. Trong thời gian làm việc với thầy, không những chúng tôi không ngừng tiếp thu thêm nhiều kiến thức chuyên môn mà còn học tập được tinh thần làm việc, thái độ làm việc nghiêm túc, tinh thần nghiên cứu khoa học cao, đây là điều rất cần thiết cho bản thân trong quá trình học tập và làm việc sau này.

Trong quá trình hoàn thành đồ án cũng như hoàn thành luận văn tốt nghiệp, do trình độ chuyên môn cũng như kinh nghiệm thực tiễn còn hạn chế nên bài luận văn không thể tránh khỏi những thiếu sót, nhóm chúng tôi rất mong các thầy cô bỏ qua và đóng góp ý kiến để chúng tôi có thể hoàn thiện bài luận văn tốt hơn.

Kính chúc quý thầy cô thật nhiều sức khỏe, hạnh phúc và thành công.

Tp. Hồ Chí Minh, ngày tháng năm 2021

Nhóm sinh viên thực hiện

TÓM TẮT

Dựa trên nhu cầu thực tế, Hệ thống máy nhận dạng nhân trắc được thiết kế nhằm giải quyết vấn đề về thời gian và hiệu suất lấy dữ liệu so với các biện pháp thủ công khác. Nâng cao khả năng làm việc liên tục, nhanh chóng, chính xác giúp giảm thiểu tối đa thời gian đo kiểm và chi phí vận hành. Cơ chế vận hành của máy quét nhận dạng nhân trắc là chụp các góc xác định trên khuôn mặt của một người, sử dụng hệ thống học sâu (deep learning) tự động nhận dạng các vùng điểm đặc trưng trên khuôn mặt từ những bức hình vừa chụp được. Từ đó tính toán ra khoảng cách, góc độ rồi xuất ra file lưu trữ. Trong bài luận này, nhóm nghiên cứu sẽ trình bày nguyên lý thiết kế, hệ thống điều khiển và thuật toán nhận dạng sử dụng trí thông minh nhân tạo để nhận dạng các vùng điểm đặc trưng trên mặt người từ đó đưa ra file dữ liệu lưu trữ làm cơ sở để nghiên cứu, ứng dụng công nghệ sau này. Nhìn chung, mục tiêu nghiên cứu thiết kế chế tạo hệ thống máy quét và xử lý nhận dạng nhân trắc (tai, mũi) đã được phát triển thành công, giải quyết được nhu cầu của thị trường đề ra.

MỤC LỤC

LỜI CẢM ƠN	i
TÓM TẮT	ii
MỤC LỤC.....	iii
DANH SÁCH CÁC HÌNH ẢNH.....	vii
DANH SÁCH CÁC BẢNG BIỂU	x
CHƯƠNG 1: GIỚI THIỆU CHUNG.	1
1.1) Đặt vấn đề	1
1.2) Nhân trắc học và nhân trắc tai – mũi.	1
1.2.1) Nhân trắc học là gì?.....	1
1.2.2) Nhân trắc học tai – mũi	1
1.3) Các phương pháp đo đặc nhân trắc có sẵn hiện nay.	1
1.3.1) Đo thủ bằng các dụng cụ đo chuyên dụng.....	1
1.3.2) Đo thông qua việc chụp ảnh và phần mềm.....	3
1.3.3) Đo thông qua việc tái tạo 3D khuôn mặt.	3
1.4) Ý nghĩa khoa học và thực tiễn của đề tài.....	4
1.5) Tình hình nghiên cứu trong và ngoài nước.....	5
1.6) Lý chọn đề tài.	5
1.7) Nhiệm vụ và yêu cầu của đề tài.	6
1.8) Đối tượng và phạm vi nghiên cứu	6
1.9) Phương pháp nghiên cứu	6
1.10) Kết cấu đồ án	6
CHƯƠNG 2: CÁC GIẢI PHÁP THIẾT KẾ VÀ THIẾT KẾ MÁY	8
2.1) Chọn hướng thiết kế.....	8
2.2) Cơ sở lý thuyết.....	9
2.2.1) Nguyên lý hoạt động	9
2.2.2) Sơ đồ động	9
2.2.3) Các phương án truyền động có thể trang bị	10
2.2.4) Các loại động cơ phổ biến có thể trang bị.....	11
2.3) Chọn thiết kế tối ưu nhất cho phương án:	13

2.4) Các tính toán cần thiết	14
2.4.1) Tính toán và lựa chọn động cơ.....	14
2.4.2) Chọn các thông số cho bộ truyền dài.....	15
2.4.3) Tính toán và thiết kế trực	15
2.4.4) Chọn ố lăn.....	17
2.4.5) Xét tính bền của các thanh chịu lực chính	18
2.4.6) Xem xét khả năng hệ bị lật	20
2.4.7) Tính toán độ bền của trụ đỡ chính	21
CHƯƠNG 3: THIẾT KẾ HỆ THỐNG ĐIỆN VÀ ĐIỀU KHIỂN.....	22
3.1) Nguyên lý hoạt động.....	22
3.1.1) Chế độ Auto	22
3.1.2) Chế độ Normal.....	22
3.1.3) Nâng cơ cầu chụp	22
3.2) Cơ cấu bộ điều khiển.....	23
3.2.1) Sơ đồ khái niệm điều khiển	23
3.2.2) Sơ đồ giải thuật.....	24
3.3) Lựa chọn các phương án điều khiển.....	25
3.3.1) Vi xử lí Arduino Mega	25
3.3.2) Driver điều khiển động cơ Servo	26
3.3.3) Modul điều khiển động cơ DC BTS 7960	29
3.3.4) Động cơ DC Servo	31
3.4.5) Động cơ DC 12/24V	33
3.4.6) Cầu dao tự động.....	34
3.4.7) Nút nhấn	35
3.4.8) Nguồn	36
3.5) Thiết kế mạch điện cho hệ thống	36
3.5.1) Sơ đồ mạch điều khiển.....	36
3.5.2) Lưu đồ điều khiển vị trí	37
3.5.3) Tính toán thiết kế bộ điều khiển PID.....	37
3.5.4) Tính toán PID.....	41
3.5.6 Lập trình cho hệ thống trên DC TunerPro	43

CHƯƠNG 4: CƠ SỞ LÍ THUYẾT VÀ THUẬT TOÁN NHÂN DẠNG NHÂN TRẮC	44
4.1) Khái niệm thuật toán liên quan.....	44
4.1.1) Deep Learning và Machine Learning.....	44
4.2) Convolutional Neural Network.....	48
4.2.1) Tổng quát.....	48
4.2.2) Lớp tích chập (Convolutional Layer).....	49
4.2.3) Lớp tổng hợp (Pooling layer).....	50
4.2.4) Lớp kết nối đầy đủ (Full connect layer).....	51
4.2.5) Batch Normalization.....	51
4.2.6) Thuật toán YOLO (You Only Look Once)	52
4.2.7) Hệ thống Grid (Grid system).....	53
4.2.8) Gán nhãn dữ liệu ảnh.....	55
4.2.9) Hàm mất mát	57
4.3) Thuật toán YOLOv4	58
4.4) Cấu trúc nhận diện vật thể của YOLOv4	59
4.5) Mô hình pre-trained:.....	66
CHƯƠNG 5: NHẬN DẠNG TÍNH TOÁN THÔNG SỐ NHÂN TRẮC.....	67
5.1) Xây dựng mô hình nhận dạng đặc điểm nhân trắc tai, mũi bằng ảnh chụp	67
5.2) Sơ đồ khái tổng quát.....	67
5.5) Xây dựng mô hình nhận dạng nhân trắc bằng YOLOv4 sử dụng Google Colab	73
5.7) Tạo dao diện GUI và kết nối với mySQL.....	75
5.7.1) Giao diện.....	75
CHƯƠNG 6 : KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ.	78
6.1) Kết quả thực nghiệm:.....	78
6.2) Đánh giá thực nghiệm	81
6.3) Đánh giá độ chính xác của lấy dữ liệu tự động và lấy dữ liệu bằng tay....	86
6.4) Tổng quan đánh giá về thiết bị	87
CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	88
7.1) Kết luận.....	88

7.2) Hướng phát triển.....	88
PHỤ LỤC 1.....	89
PHỤ LỤC 2.....	93
DANH MỤC TÀI LIỆU THAM KHẢO	98

DANH SÁCH CÁC HÌNH ẢNH

Hình 1.3.1a : Minh họa các điểm đặc trưng trên khuôn mặt	2
Hình 1.3.1b: Dụng cụ đo các điểm nhân trắc	2
Hình 1.3.2a : Đo khoảng cách điểm bằng phần mềm.....	3
Hình 1.3.2b: Phần mềm đo góc và khoảng cách các điểm nhân trắc.	3
Hình 1.3.3: Sử dụng máy chụp 3D phân tích các điểm trên khuôn mặt.....	4
Hình 1.4 : Phương pháp đo truyền thống.....	5
Hình 2.2.2: Sơ đồ động	9
Hình 2.4.2a Chiều dài trực	15
Hình 2.4.2b Các lực lên trực	16
Hình 3.4.4.3 Momen lên trực.....	17
Hình 2.4.5a Momen lên thanh ngang chính của máy.....	18
Hình 2.4.5b Momen lên thanh ngang cụm máy ảnh	19
Hình 2.4.6a Xét khả năng vật lật	20
Hình 2.4.6b Biểu đồ lực khi chiếu xuống mặt phẳng ngang	20
Hình 2.4.7a Tách hệ vật.....	21
Hình 2.4.7b Momen tổng hợp	21
Hình 3.2.1 Sơ đồ điều khiển	23
Hình 3.2.2 Sơ đồ giải thuật.....	24
Hình 3.3.1 Vi xử lí Adruino Mega.....	25
Hình 3.3.2 Driver điều khiển động cơ Servo	27
Hình 3.3.3 Modul điều khiển động cơ DC BTS 7960	29
Hình 4.3.4 Động cơ DC Servo	31
Hình 3.4.5 Động cơ DC Servo	33
Hình 3.4.6 Cầu dao tự động	34
Hình 3.4.7 Nút nhấn	35
Hình 3.4.8 Nguồn	36
Hình 3.5.1 Sơ đồ mạch điều khiển	36
Hình 3.5.2 Lưu đồ điều khiển vị trí.....	37
Hình 3.5.3a Tính toán thiết kế bộ điều khiển PID	37
Hình 3.5.3b Tính toán thiết kế bộ điều khiển PID	38

Hình 3.5.3c Tính toán thiết kế bộ điều khiển PID.....	41
Hình 3.5.4 Biểu đồ PID.....	42
Hình 4.1.1 Quan hệ giữa dữ liệu và khả năng dự đoán của thuật toán. Nguồn: Coursera	44
Hình 4.1.2 Kiến trúc mạng Neural Network.	45
Hình 4.1.3 Mối quan hệ giữa ngõ vào và ngõ ra của một node.....	46
Hình 4.1.6 Đồ thị hàm Mish	48
Hình 4.2.1 Cấu trúc mạng CNN.....	49
Hình 4.2.2. Các thông số của conv layers	50
Hình 4.2.3 Thuật toán max-pooling.....	51
Hình 4.2.4 Chuyển lớp tổng hợp về một vector.....	51
Hình 4.2.7a Tính bảo toàn vị trí và đặc tính của vật khi sử dụng hệ thống Grid	53
Hình 4.2.7b : Mô tả ma trận grid.....	54
Hình 4.2.7c : Tính toán tỉ lệ IoU	55
Hình 4.2.8: Tâm của 2 vật thể cùng nằm trong 1 grid.	57
Hình 5.2.9: Thuật toán được áp dụng ở ngõ ra.....	58
Hình 4.3 So sánh tốc độ xử lý và độ chính xác của YOLOv4 với các thuật toán khác trong cùng tập dữ liệu MS COCO.....	59
Hình 4.4.1: Cấu trúc nhận diện vật thể của YOLOv4	60
Hình 4.4.2: Sự khác biệt giữa mạng DenseNet và mạng CSPDenseNet.....	60
Hình 4.4.3: Quá trình lấy Batch-Norm, ReLU, tích chập với filter 3x3 của 1 lớp	61
Hình 4.4.4 Quá trình ghép chồng lớp tích chập phía trước vào lớp tích chập phía sau	61
Hình 4.4.5 Mô hình DenseNet với 3 khối Dense Block và các lớp chuyển chèn giữa.....	62
Hình 4.4.6 Mô hình tích chập Darknet-53 gồm 53 tầng tích chập.....	62
Hình 4.4.7 : Mô hình FPN.....	63
Hình 4.4.8: Mô hình YOLO-SPP	64
Hình 4.4.9: Mô hình mạng PAN	64
Hình 4.4.10: Thay vì thực hiện cộng như mô hình PAN, YOLOv4 thực hiện phép gộp.....	65

Hình 4.4.11 ROI sẽ đi qua mạng SPP để thu được prediction	65
Hình 5.2a Một số hình ảnh minh họa trong tập dữ liệu train	69
Hình 5.2b Vị trí các điểm nhân trắc trên khuôn mặt theo hai hướng đối diện và bên phải.....	70
Hình 5.3 Mô tả qua trình histogram hóa.....	72
Hình 5.4.1 : Kết quả nhận diện các điểm nhân trắc trên mặt (hướng chính diện).....	74
Hình 5.4.2 : Xác định số pixel ứng với 1 cm trên ảnh.	74
Hình 5.5.1 Giao diện phần mềm.	76
Hình 5.5.2 Đưa data dữ liệu lên SQL.....	77
Hình 5.5.3 Lưu dữ liệu vừa tính được lên database.....	77
Hình 6.1.1 : Kết quả nhận diện các điểm nhân trắc trên mặt (hướng chính diện).....	79
Hình 6.1.2: Kết quả nhận diện các điểm nhân trắc tai và mũi (tai phải).....	79
Hình 6.1.3: Kết quả nhận diện các điểm nhân trắc tai và mũi (tai trái)	80
Hình 6.1.4 : Tọa độ góc và khoảng cách được lưu trữ trên file .csv	80
Hình 6.3.1 Đánh giá độ chính xác của dữ liệu	86
Hình 5.6.5 : Thay đổi runtime type của Google Colab để sử dụng GPU	91
Hình 5.6.6 : Liên kết Google Drive với Google Colab.....	91

DANH SÁCH CÁC BẢNG BIỂU

BẢNG 2.3: SO SÁNH CÁC PHƯƠNG PHÁP NHẬN DẠNG NHÂN TRẮC.....	14
BẢNG 3.3 SO SÁNH CÁC PHƯƠNG ÁN THIẾT KẾ	20
BẢNG 3.3.2.3 LỆNH ĐIỀU KHIỂN.....	35
BẢNG 5.1.2.1 CÁC ÚNG DỤNG KHÁC NHAU CHÚNG TA CÓ THỂ ÁP DỤNG CÁC NEURON NETWORK KHÁC NHAU.....	53
BẢNG 5.6.1 : TÊN LABEL VÀ VỊ TRÍ CỦA CÁC ĐIỂM NHÂN TRẮC CẦN NHẬN DẠNG.	77
BẢNG 6.2.1: ĐO NHÂN TRẮC QUA ẢNH KỸ THUẬT SỐ	88
BẢNG 6.2: ĐỘ CHÍNH XÁC THUẬT TOÁN TRONG QUÁ TRÌNH XÂY DỰNG MODEL.....	90

BẢNG KÍ HIỆU SỬ DỤNG TRONG ĐỒ ÁN

Kí hiệu	Ý nghĩa	Kí hiệu	Ý nghĩa
s	Stride: Số bước trượt trong một tích chập	y	Scaling factor hay scaling parameter của layer
p	Padding: Thêm hàng,cột trong một lớp tích chập	β	Shiftng factor hay offset của layer
n_c	Số filter được sử dụng ở mỗi lớp tích chập	ϵ	Smoothing factor để tránh việc chia 0 làm vô nghĩa, thường có giá trị rất nhỏ
$f*f$	Kích thước của filter	y_i	Đầu ra của batch normalization
H	Chiều dài của lớp tích chập	b_w	Chiều rộng của box
W	Chiều rộng của lớp tích chập	c_i	Xác suất xuất hiện của vật thể thứ i trong grid
D	Số chiều của lớp tích chập	S^2	Tổng số grid trong 1 ảnh.
p_c	Xác suất grid đang xét chứa vật thể	i	Mẫu thứ i
b_x	Tọa độ x của góc phía trên bên trái của box	B	Số box
b_y	Tọa độ y của góc phía trên bên dưới của box	σ_B	Độ lệch chuẩn thực nghiệm cũng được tính toán trong mini-batch B

b_h	Chiều cao của box	m	Số lượng instance trong mini-batch B
μ_B	Giá trị kì vọng thực nghiệm được tính toán trên mini-batch B	\hat{x}_i	Giá trị đầu vào thứ i trong mini-batch sau khi chuẩn hóa và tâm không
σ_B	Độ lệch chuẩn thực nghiệm cũng được tính toán trong mini-batch B		

CHƯƠNG 1: GIỚI THIỆU CHUNG.

Chương này giới thiệu sơ bộ về nhân trắc học, tình hình nghiên cứu, lấy dữ liệu nhân trắc hiện nay trong và ngoài nước. Từ đó đưa ra lý do chọn đề tài và tóm tắt quá trình thực hiện đề tài.

1.1) Đặt vấn đề.

Tái cấu trúc tai – mũi là việc thay đổi cấu trúc tai – mũi ban đầu hoặc đưa về hình dạng tự nhiên bằng việc phẫu thuật tạo hình hoặc dùng bộ phận giả. Để có thể tái cấu trúc một phần nào đó trên cơ thể ta cần biết những điểm đặc trưng, trọng yếu tạo nên cấu trúc của bộ phận đó.

Vì vậy việc xác định các điểm nhân trắc, nhất là vùng thẩm mỹ (mắt, tai, mũi miệng) đặc biệt quan trọng và cấp thiết. Những dữ liệu được nhận dạng sẽ là cơ sở để tính toán ứng dụng các công nghệ sau này.

Trước xu thế cơ khí hóa, tự động hóa cùng việc ứng dụng trí tuệ nhân tạo (AI) vào tất cả lĩnh vực trong đời sống hiện nay. Trong những năm gần đây, ngành y tế đã đạt được những thành tựu vượt bậc nhờ việc áp dụng cách thành tựu công nghệ. Trong đó, kỹ thuật y sinh, một lĩnh vực mà việc sử dụng các máy móc thiết bị và công nghệ thông tin là 1 phần không thể thiếu. Cụ thể, 1 lĩnh vực được ứng dụng nhiều nhất trong cuộc sống là nhận dạng các điểm cấu trúc trên cơ thể người. Ở đây, chúng tôi chỉ tập trung nghiên cứu về 2 bộ phận đó là: mũi và tai.

1.2) Nhân trắc học và nhân trắc tai – mũi.

1.2.1) Nhân trắc học là gì?

Nhân trắc học tên khoa học là Écgônômi là khoa học nghiên cứu sự cấu thành, kích thước, tỷ lệ và mối quan hệ các bộ phận trên cơ thể con người, từ đó đưa ra sự phù hợp trong công việc, hệ thống máy móc, thiết bị, sản phẩm, môi trường với các khả năng về thể lực, trí tuệ và cả với những hạn chế của con người. Từ Ecgonomi xuất phát từ tiếng Hy Lạp: Ergo – Lao động; Nomos – Quy luật. Ở một số nước khái niệm Écgônômi đồng nghĩa với khái niệm các yếu tố con người.

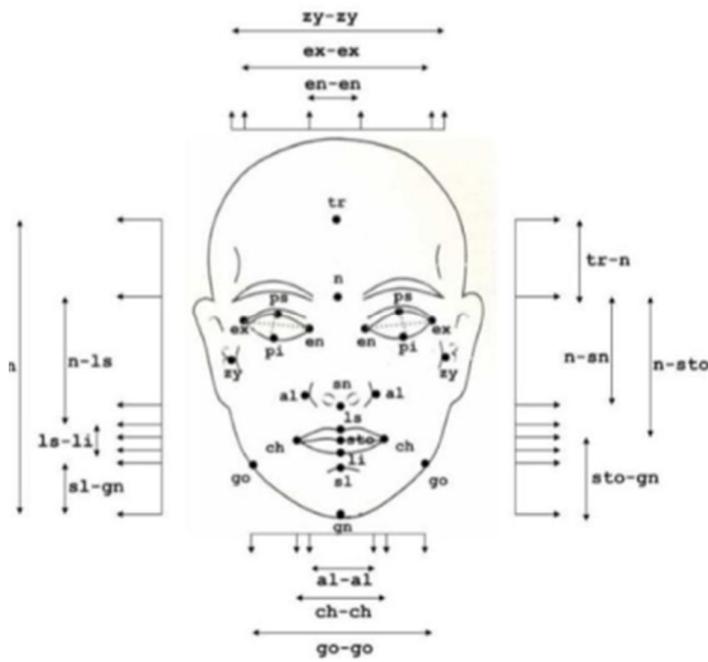
1.2.2) Nhân trắc học tai – mũi

Bao gồm các điểm và các kích thước được qui định trong “BẢNG 1.1: ĐO NHÂN TRẮC QUA ẢNH KỸ THUẬT SỐ” PHỤ LỤC 2.

1.3) Các phương pháp đo đặc nhân trắc có sẵn hiện nay.

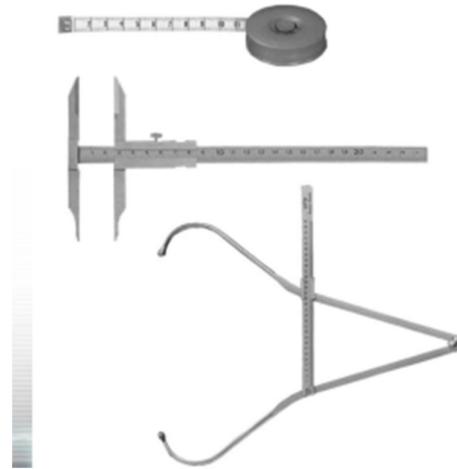
1.3.1) Đo thủ bằng các dụng cụ đo chuyên dụng.

- Có nhiều quy ước xác định các mốc nhân trắc trên khuôn mặt, ban đầu có 220 điểm do Walker và Kowalski đánh dấu và quy định, sau đó Farkas đã cải tiến việc đánh dấu lại bằng cách chỉ đánh 100 vùng đặc trưng trên khuôn mặt. Về sau Jorgensen đã chọn lại 30 điểm mốc từ Farkas.



Hình 1.3.1a : Minh họa các điểm đặc trưng trên khuôn mặt.

- Việc đánh dấu các điểm nhân trắc và tiến hành đo bằng phương pháp thủ công khá tốn thời gian, nhưng mang lại hiệu quả không hề cao, nhát là ở vùng mắt, hốc mũi, vv. Việc lựa chọn thay đổi thiết bị đo chuyên dụng cho từng vùng trên bề mặt mắt, tai, mũi kién cho công việc lấy dữ liệu đã phức tạp nay còn phức tạp hơn. Bên cạnh đó việc sử dụng chung thiết bị đo cho nhiều vùng nhạy cảm như mắt, mũi miệng có nguy cơ lây nhiễm nguồn bệnh không mong muốn, mỗi lần đo song một người lại phải tiến hành sát khuẩn.



Hình 1.3.1b: Dụng cụ đo các điểm nhân trắc.

1.3.2) Đo thông qua việc chụp ảnh và phần mềm.

- Hiện nay bằng công nghệ hiện đại đã có các phương pháp khác tiên tiến hơn như “Phân tích các vùng mặt trên ảnh”[4], và chụp ảnh chuẩn hóa máy chụp 3D.



Hình 1.3.2a : Đo khoảng cách điểm bằng phần mềm.

- Nhìn qua công nghệ chụp ảnh có điểm mạnh là không cần sử dụng các công cụ đo như thường lệ, tiết kiệm rất nhiều thời gian. Nhưng đa số sau khi có được dữ liệu các điểm, họ phải sử dụng phần mềm và đo khoảng cách bằng tay.



Hình 1.3.2b: Phần mềm đo góc và khoảng cách các điểm nhân trắc.

1.3.3) Đo thông qua việc tái tạo 3D khuôn mặt.

- Với công nghệ chụp 3D, kết quả đo sẽ chính xác hơn do ảnh ít bị biến dạng hơn trong quá trình chụp. Nhưng kèm theo đó là giá thành khá mắc, sau khi có hình 3D, chúng ta phải sử dụng phần mềm để đo bằng tay như ở phương pháp “Phân tích vùng ảnh”.



Hình 1.3.3: Sử dụng máy chụp 3D phân tích các điểm trên khuôn mặt.

1.4) Ý nghĩa khoa học và thực tiễn của đè tài.

❖ Ý nghĩa khoa học:

- Tạo điều kiện, tiền đề cho người nghiên cứu phát triển các công nghệ mới, vận dụng kiến thức đã học được vào thực tế để nâng cao chất lượng cuộc sống xã hội, cụ thể là cải thiện kết quả chuẩn đoán thông qua máy móc và ứng dụng trí tuệ nhân tạo (machine learning).
- Tạo ra sản phẩm cải tiến về hình dáng và công nghệ, cụ thể là máy nhân trắc với chi phí thấp và tính năng công nghệ cao.
- Tạo tiền đề để cải tiến, phát triển các sản phẩm khác và ứng dụng machine learning vào trong nhiều lĩnh vực đời sống xã hội.

❖ Ý nghĩa thực tiễn:

- Trong thời gian tiếp cận quy trình đo chụp nhân trắc của bên đặt hàng, chúng tôi nhận thấy rằng, quy trình đo đạc các điểm trên khuôn mặt hết sức khó khăn và bất tiện. Bên cạnh đó, các phép đo truyền thống thường có sai số rất lớn vì không thể tiếp cận các vùng nhạy cảm, dễ tổn thương trên khuôn mặt.



Hình 1.4 : Phương pháp đo truyền thống.

- Qua đó nhóm đã đề xuất phương án thiết kế chế tạo máy có thể đo kiểm tự động, nhằm tăng năng suất, độ chính xác so với phương pháp truyền thống. Đây nhanh quá trình lấy số liệu, phục vụ công tác nghiên cứu và ứng dụng.

1.5) Tình hình nghiên cứu trong và ngoài nước.

❖ Trong nước:

- Nước ta đang trên đà phát triển rất nhanh đặc biệt là lĩnh vực y tế, vì vậy một số bệnh viện lớn như chợ rẫy, đại học y dược,... đã có những phần mềm chuyên biệt để đo gián tiếp nhân trắc người sử dụng ảnh chụp. Và phần còn lại đa số vẫn khảo sát và ghi nhận dữ liệu bằng tay vì chi phí phần mềm và công cụ đo kiểm tự động khá đắt đỏ.

❖ Quốc tế:

- Lĩnh vực nghiên cứu các điểm nhân trắc vùng khuôn mặt luôn là một chủ đề nóng hổi mà các nước phát triển quan tâm. Nó là tiền đề để phát triển các phương pháp phẫu thuật hay tái cấu trúc các bộ phận, tùy vào nhiều mục đích khác nhau. Phương pháp thường dùng là đo gián tiếp nhân trắc bằng ảnh chụp, một số bệnh viện, phòng nghiên cứu nổi tiếng ở Mỹ đã áp dụng việc quét hình ảnh 3D vào lấy dữ liệu nhân trắc. Với độ chính xác khá cao.

1.6 Lý chọn đề tài.

Hiện nay, việc tái cấu trúc 2 bộ phận mũi và tai được thực hiện thông qua việc tư vấn của các bác sĩ có chuyên môn, thông qua việc đo đạc số liệu bằng dụng cụ đo thủ công (thước kẹp, thước đo góc,...). Từ đó, đưa ra các phương án thích hợp nhất cho việc tái cấu trúc. Công việc này tốn nhiều thời gian và công sức nhưng hiệu quả mang lại không nhiều, độ sai lệch lớn. Do đó, để có độ chính xác tin cậy, ta phải đo lại nhiều lần, dẫn đến việc phải chi nhiều thời gian và tiền bạc. Nhưng với sự phát triển công nghệ và kỹ thuật hiện nay việc tự động hóa quá trình đo đạc, so sánh các

thông số nhân trắc, thì việc khuôn mặt sau khi tái cấu trúc tai – mũi đạt được tỉ lệ “vàng” là điều hoàn toàn có thể thực hiện.

Năm bắt được xu thế đó nhóm chúng tôi quyết định thực hiện đề tài tài “Nghiên cứu, thiết kế và chế tạo máy và xử lý nhận dạng nhân trắc (tai-mũi)”. Với mong muốn giúp các bác sĩ, các nhóm nghiên cứu về nhân trắc học tiết kiệm thời gian, công sức đo đạc, các thông số trên khuôn mặt, cụ thể ở đây là (tai, mũi).

1.7) Nhiệm vụ và yêu cầu của đề tài.

❖ Nhiệm vụ:

- Phân tích, đánh giá thực trạng và những giải pháp có thể dùng để nhận diện các đặc điểm nhân trắc.
- Lên phương án thiết kế, chế tạo máy và hệ thống điều khiển.
- Thiết kế chế tạo khung máy, lập trình điều khiển, ứng dụng mạng deep learning để nhận dạng, tính toán các số liệu nhân trắc.
- Chế tạo hoàn thiện máy, chạy thực nghiệm, đánh giá và vận hành.

❖ Yêu cầu:

- Nguyên lý máy.
- Bản vẽ thiết kế.
- Phần mềm lập trình
- Chế tạo.
- Lắp ráp.

1.8) Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu: Đặc điểm nhân trắc 2 bộ phận tai - mũi tại Việt Nam nói riêng và trên thế giới nói chung, các giải pháp nhận diện và thiết kế máy.
- Phạm vi nghiên cứu: Phạm vi nghiên cứu là công dân Việt Nam độ tuổi từ 19-50.

1.9) Phương pháp nghiên cứu

- Phương pháp phân tích và đánh giá nhu cầu cần thiết về sản phẩm.
- Phương pháp thu thập và so sánh để chọn phương án thực hiện tối ưu.
- Phương pháp xây dựng nguyên lý máy.
- Phương pháp chế tạo.

1.10) Kết cấu đồ án

Chương 1: Giới thiệu chung.

Chương 2: Các giải pháp thiết kế và thiết kế máy

Chương 3: Thiết kế hệ thống điện và điều khiển

Chương 4: Cơ sở lý thuyết và thuật toán nhận dạng nhân trắc.

Chương 5: Nhận dạng, tính toán thông số nhân trắc.

Chương 6: Kết quả thực nghiệm và đánh giá
Chương 7: Kết luận và hướng phát triển

CHƯƠNG 2: CÁC GIẢI PHÁP THIẾT KẾ VÀ THIẾT KẾ MÁY

Ở chương này, nhóm nghiên cứu đã đưa ra nhiều phương án khả thi để chế tạo máy nhận dạng nhân trắc. Từ đó chọn ra phương pháp tối ưu nhất đưa vào tính toán thiết kế, chế tạo.

2.1) Chọn hướng thiết kế.

Nhóm chúng tôi quyết định chọn phương pháp nhận diện và đo đạc các chỉ số nhân trắc tai – mũi thông qua ảnh chụp vì các lí do sau:

- ❖ Với sự phát triển ngày càng lớn của trí tuệ nhân tạo, việc nhận diện các điểm nhân trắc của tai – mũi một cách chính xác với độ sai số thấp là việc hoàn toàn có thể, từ đó việc tính toán các kích thước, các góc diễn ra dễ dàng.
 - ❖ Giá thành từ 5-10 triệu phù hợp với kinh phí cho 1 đồ án tốt nghiệp.
- Với chứng năng chính của máy là: chụp đủ các mặt tai – mũi trên khuôn mặt
- Dưới đây là các phương án cũng như các tiêu chí để so sánh
- ❖ Phương án 1: Có định người chụp trong tư thế đứng, camera quay xung quanh
 - Khả năng chụp đủ các mặt tai, mũi: Có
 - Hệ tọa độ thay đổi trong quá trình chụp: Không
 - Tổng thời gian chụp (ước tính): 1 phút
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 2m
 - ❖ Phương án 2: Có định Camera, xoay người trong tư thế đứng
 - Khả năng chụp đủ các mặt tai, mũi: Có
 - Hệ tọa độ thay đổi trong quá trình chụp: Có
 - Tổng thời gian chụp (ước tính): 1 phút 20 giây
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 2m
 - ❖ Phương án 3: Kết hợp xoay người và xoay camera trong tư thế đứng
 - Khả năng chụp đủ các mặt tai, mũi: Có
 - Hệ tọa độ thay đổi trong quá trình chụp: Có
 - Tổng thời gian chụp (ước tính): 1 phút
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 2m
 - ❖ Phương án 4: Có định người ngồi, camera xoay xung quanh
 - Khả năng chụp đủ các mặt tai, mũi: Có
 - Hệ tọa độ thay đổi trong quá trình chụp: Không
 - Tổng thời gian chụp (ước tính): 1 phút
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 1m
 - ❖ Phương án 5: Camera đứng yên xoay người trong tư thế ngồi
 - Khả năng chụp đủ các mặt tai, mũi: Có

- Hệ tọa độ thay đổi trong quá trình chụp: Có
 - Tổng thời gian chụp (ước tính): 1 phút 20 giây
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 1m4
 - ❖ Phương án 6: Kết hợp Xoay Người trong tư thế nằm, xoay camera xung quanh
 - Khả năng chụp đủ các mặt tai, mũi: Có
 - Hệ tọa độ thay đổi trong quá trình chụp: Có
 - Tổng thời gian chụp (ước tính): 2 phút
 - Kích thước lớn nhất của cơ cấu trong không gian (ước tính): 2m
- => Sau khi so sánh các tiêu chí, nhóm chúng em chọn Phương án 4.

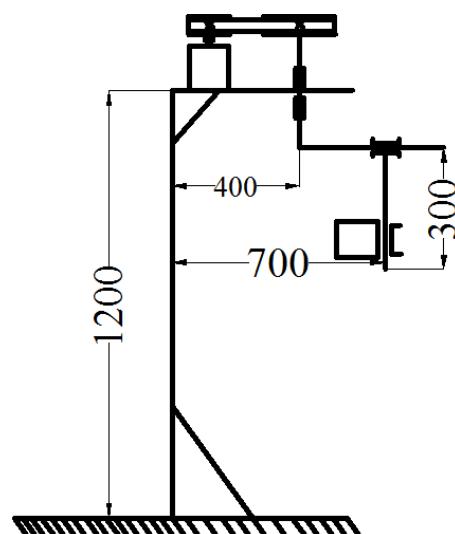
2.2) Cơ sở lý thuyết.

Ở phần này chúng ta sẽ tìm hiểu cơ sở lý thuyết về nguyên lý hoạt động của máy, các phương án truyền động, các phương pháp điều khiển, xử lý ảnh và nhận dạng. Qua đó, tạo cơ sở để chọn lựa thiết kế tối ưu

2.2.1) Nguyên lý hoạt động

Momen từ động cơ được truyền qua bộ truyền đến cụm lắp máy ảnh, người chụp sẽ được ngồi trên ghế và trực cơ thể sẽ trùng với trực quay của cụm. Khi xoay, sẽ có những khoảng thời gian động cơ sẽ dừng lại để chụp ảnh, số lần dừng lại sẽ là 4 và thời gian từ bức ảnh này đến bức ảnh kế tiếp là 9 giây.

2.2.2) Sơ đồ động



Hình 2.2.2: Sơ đồ động

2.2.3) Các phương án truyền động có thể trang bị

a) Truyền động đai

Truyền động đai là truyền chuyển động và công suất từ bánh đai dẫn đến bánh đai bị dẫn nhờ lực ma sát giữa đai và các bánh đai. Bộ truyền làm việc theo nguyên lý ma sát. Lực ma sát được tạo ra bởi lực căng ban đầu trên các nhánh đai.

Cấu tạo: bánh đai, dây đai, có thể có bánh căng hoặc bánh dẫn hướng đai.

Phân loại:

- Theo hình dạng của tiết diện đai: đai tròn, đai dẹt, đai thang, đai hình lược (đai nhiều chẽm), đai răng.
- Theo vị trí của các trục và chiều quay của các bánh đai: truyền động thường, truyền động chéo, truyền động nửa góc, truyền động góc.

❖ Ưu điểm

- Có khả năng truyền động giữa các trục xa nhau.
- Làm việc êm.
- Kết cấu đơn giản.
- Đề phòng quá tải cho máy.
- Phí tổn bảo dưỡng ít.

❖ Nhược điểm

- Tỷ số truyền không phải là hằng số.
- Lực tác dụng lên trục và ô trục lớn.
- Bị trượt qua sự giãn nở của dây đai.
- Nhiệt độ ứng dụng bị giới hạn.

b) Truyền động bánh răng

Truyền động bánh răng được thực hiện truyền động quay nhờ vào sự ăn khớp giữa các răng trên hai bánh răng, dạng truyền động này dùng thể thay đổi vận tốc, momen và chiều quay.

❖ Ưu điểm:

- Kích thước nhỏ gọn hơn các bộ truyền khác.
- Khả năng tải cao hơn so với các bộ truyền khác có cùng kích thước.
- Tỷ số truyền không thay đổi, số vòng quay n_2 ổn định.
- Hiệu suất truyền động cao hơn các bộ truyền khác.
- Làm việc chắc chắn, tin cậy có tuổi bền cao.

❖ Nhược điểm:

- Yêu cầu gia công có độ chính xác cao, giá thành tương đối đắt.
- Khi làm việc có nhiều tiếng ồn, nhất là khi vận tốc làm việc cao.

- Cần bảo dưỡng thường xuyên.

2.2.4) Các loại động cơ phổ biến có thể trang bị

a) Động cơ giảm tốc

Động cơ giảm tốc được định nghĩa là động cơ điện có tốc độ thấp, tốc độ đã giảm đi nhiều (có thể là 1/2, 1/3, 1/4, 1/5, 1/8, 1/10, 1/15...) so với động cơ thông thường ở cùng công suất và số cực.

Động cơ giảm tốc được cấu tạo từ hai thành phần chính:

- Động cơ điện.
- Hộp giảm tốc.

Động cơ điện lại có cấu tạo gồm 2 phần chính đó là phần tĩnh Stato và phần động Roto. Hộp giảm tốc bên trong chứa bộ truyền động sử dụng bánh răng, trục vít... để làm giảm tốc độ vòng quay, tăng momen xoắn và là bộ phận trung gian giữa động cơ điện với bộ phận làm việc của máy công tác. Đầu còn lại của hộp giảm tốc nối với tải.

b) Động cơ bước

Động cơ bước là một động cơ đồng bộ dùng để biến đổi các tín hiệu điều khiển dưới dạng các xung điện rời rạc kế tiếp nhau thành các chuyển động góc quay hoặc các chuyển động của roto và có khả năng cố định roto vào những vị trí cần thiết.

- ❖ Phân loại động cơ bước theo số pha (số cuộn dây).
- ❖ Phân loại động cơ bước theo cực:
 - Động cơ đơn cực: dòng điện trong một động cơ đơn cực luôn chạy qua cuộn dây theo cùng một hướng, cho phép sử dụng mạch điều khiển đơn giản, tạo ra mô-men xoắn ít hơn động cơ lưỡng cực.
 - Động cơ lưỡng cực: dòng điện trong một động cơ lưỡng cực có thể chạy qua cuộn dây theo một trong hai hướng, đòi hỏi một mạch điều khiển phức tạp hơn động cơ đơn cực, tạo ra nhiều mô-men xoắn hơn.
- ❖ Phân loại động cơ bước theo Rotor:
 - Động cơ bước nam châm vĩnh cửu (Permanent Magnet Stepper): sử dụng một nam châm vĩnh cửu trong rotor và hoạt động dựa trên lực hút hoặc lực đẩy giữa rotor và nam châm điện stator.
 - Động cơ bước biến đổi điện trở (Variable Reluctance Stepper): có một rotor sắt tròn và hoạt động dựa trên nguyên tắc miễn cưỡng tối thiểu xảy ra với khe hở tối thiểu, do đó các điểm rotor bị hút về phía cực nam châm của stator.
 - Động cơ bước đồng bộ lai (Hybrid Synchronous Stepper): kết hợp các kỹ thuật nam châm vĩnh cửu (PM) và biến đổi điện trở (VR) để đạt được công suất tối đa trong kích thước nhỏ gọn.

c) Động cơ servo

Động cơ servo là một thiết bị điện từ sử dụng cơ chế phản hồi âm để chuyển đổi tín hiệu điện thành chuyển động có kiểm soát. Về cơ bản, các động cơ servo hoạt động giống như các bộ truyền động cung cấp khả năng kiểm soát chính xác về vận tốc, gia tốc và vị trí.

Chủ yếu phân làm 3 loại:

- Động cơ servo DC dựa trên nền tảng động.
- Động cơ servo AC đồng bộ.
- Động cơ servo AC cảm ứng.

Cấu tạo của một động cơ Servo gồm 4 thành phần chính:

- Động cơ một chiều
- Cảm biến vị trí
- Bộ truyền bánh răng
- Mạch điều khiển

Cơ cấu bánh răng được kết nối với động cơ cung cấp phản hồi cho cảm biến vị trí.

Động cơ servo là thiết bị được điều khiển bằng chu trình kín. Từ tín hiệu hồi tiếp vận tốc/vị trí, hệ thống điều khiển sẽ điều khiển hoạt động của một động cơ servo. Với lý do nêu trên nên sensor đo vị trí hoặc tốc độ là các bộ phận cần thiết phải tích hợp cho một động cơ servo. Đặc tính vận hành của một động cơ servo phụ thuộc rất nhiều vào đặc tính từ và phương pháp điều khiển động cơ servo.

Rotor của động cơ là một nam châm vĩnh cửu có từ trường mạnh và stator của động cơ được cuốn các cuộn dây riêng biệt, được cấp nguồn theo một trình tự thích hợp để quay rotor.

Nếu thời điểm và dòng điện cấp tới các cuộn dây là chuẩn xác thì chuyển động quay của rotor phụ thuộc vào tần số và pha, phân cực và dòng điện chạy trong cuộn dây stator.

Động cơ servo được hình thành bởi những hệ thống hồi tiếp vòng kín. Tín hiệu ra của động cơ được nối với một mạch điều khiển. Khi động cơ vận hành thì vận tốc và vị trí sẽ được hồi tiếp về mạch điều khiển này. Khi đó bất kỳ lý do nào ngăn cản chuyển động quay của động cơ, cơ cấu hồi tiếp sẽ nhận thấy tín hiệu ra chưa đạt được vị trí mong muốn. Mạch điều khiển tiếp tục chỉnh sai lệch cho động cơ đạt được điểm chính xác nhất.

2.3) Chọn thiết kế tối ưu nhất cho phương án:

BẢNG 2.3 SO SÁNH CÁC PHƯƠNG ÁN THIẾT KẾ

Tiêu chí \ Ý tưởng	1	2	3	4	5
Hình ảnh tổng thể					
Quay quanh 1 vật	Hệ thống di chuyển trên thanh ray vòng cung	Hệ thống di chuyển trên thanh ray vòng cung	Truyền momen quay từ trục đến hệ thống thanh chữ L	Hệ thống di chuyển trên thanh ray vòng cung	Truyền momen quay từ trục đến hệ thống thanh chữ L
Tịnh tiến theo phương ngang	Không có	Thanh ray trượt bi	Thanh ray trượt bi	Dùng trực vít	Điều chỉnh trước
Tịnh tiến theo phương đứng	Dùng trực vít	Thanh ray trượt bi	Thanh ray trượt bi	Thanh ray trượt bi	Điều chỉnh trước
Chân đế	Thanh ray trượt bi	Thanh ray trượt bi	4 chân trụ	2 chân trụ	1 trụ 3 chân
Tính thẩm mỹ	Khá	Trung bình	Trung bình	Trung bình	Tốt
Bộ truyền	Không	Không	Đai	Xích	Đai

Tiêu chí \ Ý tưởng	1	2	3	4	5
Vật liệu làm trụ đỡ chính	Thép ống	Thép ống	Thép hộp	Thép tròn	Thép hộp
Khả năng vận chuyển	Trung bình	Trung bình	Thấp	Thấp	Cao
Khả năng tháo lắp	Trung bình	Trung bình	Thấp	Thấp	Cao
Hoạt động ổn định	Tốt	Tốt	Khá	Khá	Tốt
Bảo trì bảo dưỡng	Trung bình	Trung bình	Trung bình	Trung bình	Dễ dàng
Số motor sử dụng	2	2	3	2	2
Chi phí	Cao	Trung bình	Cao	Trung bình	Trung bình
Không gian của hệ thống (chiều ngang)	1m ²	1m ²	2m ²	2m ²	1m ²

=> Sau khi lập bảng so sánh các thiết kế từ phương án đã chọn. Nhóm chúng tôi chọn thiết kế 5 cho hệ thống máy của mình.

Đôi điều về lí do chọn thiết kế 5:

- Thiết kế với khả năng gấp gọn giúp tối ưu không gian.
- Bộ truyền đai răng giúp truyền động êm, vừa đảm bảo được tỉ số truyền, vừa chống quá tải.
- Không gian làm việc theo phương ngang tối ưu nhất (1m²).
- Với thép hộp việc gia công sẽ dễ dàng và tận dụng được vật liệu có sẵn.
- Dễ dàng bảo trì, sửa chữa.

2.4) Các tính toán cần thiết

2.4.1) Tính toán và lựa chọn động cơ

a) Động cơ xoay cụm máy ảnh và thước

Tải trọng = Khối lượng của cụm là: 3,5 kg

Chọn tỉ số truyền 1:2 => Ứng với bộ puly răng 30 và 60 răng

Bán kính trục động cơ là $r = 1,2$ cm. Momen tối thiểu mà động cơ cần cung cấp là:

$$M = F \times r = 3,5 \times 10 \times 0.012 = 0.42 (N.m)$$

Vận tốc cần đáp ứng là $\omega = 1$ vòng/phút

→ Vận tốc dài là

$$v = \frac{\omega \times 2\pi r}{60} = 0.075 \left(\frac{m}{s} \right)$$

Giả sử $F_{ms} = 20N$. Công suất làm việc là: $P_{lv} = F \times v = (3,5 \times 10 + 20) \times 0.075 = 4,2 (W)$

Với $\eta = 0,96$ Hiệu suất bộ truyền đai.

→ Công suất động cơ cần tối thiểu là:

$$P = \frac{P_{lv}}{\eta} = 4,4 (W)$$

→ Ta chọn động cơ DC Servo JGB37-545 loại 12VDC 37RPM với:

- Tỉ số truyền 168:1 (động cơ quay 168 vòng trục chính hộp giảm tốc quay 1 vòng).
- Dòng không tải: 200mA
- Dòng chịu đựng tối đa khi có tải: 5A
- Tốc độ không tải: 37RPM (37 vòng 1 phút)
- Tốc độ chịu đựng tối đa khi có tải: 30RPM (30 vòng 1 phút)
- Lực kéo Moment định mức: 21KG.CM
- Lực leo Moment tối đa: 84KG.CM
- Chiều dài hộp số L: 26.5mm
- Số xung Encoder mỗi kênh trên 1 vòng quay trục chính: $11 \times 168 = 1848$ xung.

b) Động cơ xoay trục 2 tầng

Tải trọng = Khối lượng của cụm là: 3,2 kg

Chọn tỉ số truyền 2:1 => Ứng với bộ puly răng 60 và 30 răng

Bán kính trục động cơ là $r = 1,4$ cm. Momen tối thiểu mà động cơ cần cung cấp là:

$$M = F \times r = 3,2 \times 10 \times 0.014 = 0.45 (N.m)$$

Vận tốc cần đáp ứng là $\omega = 120$ vòng/phút

→ Vận tốc dài là:

$$v = \frac{\omega \times 2\pi r}{60} = 0.25 (m/s)$$

Giả sử $F_{ms} = 90N$. Công suất làm việc là: $P_{lv} = F \times v = (3,2 \times 10 + 90) \times 0.25 = 30,5 (W)$

Với $\eta = 0,96$ Hiệu suất bộ truyền đai.

→ Công suất động cơ cần tối thiểu là:

$$P = \frac{P_{lv}}{\eta} = 32(W)$$

→ Ta chọn động cơ 2 DC 30w 24 GU 24 – 10K.

2.4.2) Chọn các thông số cho bộ truyền đai

Với các thông số trên ta có các lựa chọn như sau:

- Môđun và bỉ rộng đai được xác định:

+ Chọn modun $m = 3$ với cụm máy ảnh và $m = 2$ với bộ gắn trực vít

+ Bề rộng đai: $b = 6 mm$ với cụm máy ảnh (1) và $b = 10mm$ với bộ gắn trực vít (2)

- Số răng bánh đai nhỏ: 30; bánh đai lớn: 60

- Khoảng cách trực: Chọn $a = 100mm$

- Lực tác dụng lên trực: $F_r = 1000P_1/v = 58 N$ (1) và 152 N (2)

2.4.3) Tính toán và thiết kế trực

Vì trực chịu tải thấp nên ta dùng thép CT5 để thiết kế trực

Momen trên trực là:

$$T_1 = \frac{9,55 \cdot 10^6 P_1}{n_1} = \frac{9,55 \cdot 10^6 \times 4,4}{4 \times 1000} = 6844,16(N/mm)$$

Xác định sơ bộ đường kính trực được xác định theo công thức:

$$d \geq \sqrt[3]{\frac{T}{0,2[\tau]}} = 12,88mm$$

chọn $[\tau] = 15 MPa$

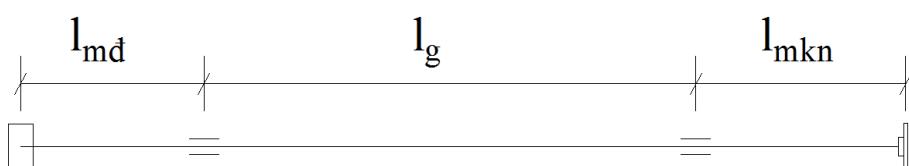
Chọn $d = 13mm \Rightarrow$ Chiều rộng ô lăn $b_0 = 13mm$

Tính toán chiều dài trực

Chiều dài may o bánh đai: $l_{md} = 1,4 \cdot 20 = 28 (mm)$

Chiều dài may o khớp nối (nối trực đòn hồi)

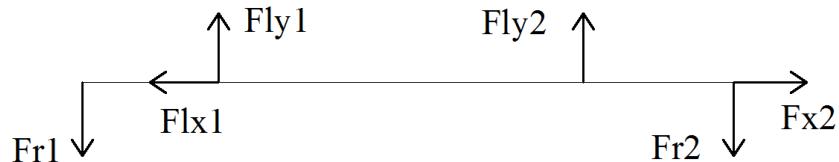
$L_{mkn} = 1,6 \cdot 20 = 32 (mm)$



Hình 2.4.2a Chiều dài trực

Chiều dài trực: $l = l_{\text{mđ}} + L_{\text{mkn}} + L_g = 28+32+75 = 135\text{mm}$

Tính toán các lực trên trực:



Hình 2.4.2b Các lực lên trực

$$\sum m_{BX} = 0 \Leftrightarrow F_{lx1} - F_{x2} = 0 \Rightarrow F_{lx1} = 31 \sin(45) = 22\text{N}$$

$$\sum m_{By} = 0 \Leftrightarrow F_{ly2}.75 - F_{r2}.107 + F_{r1}.28 = 0 \Rightarrow F_{ly2} = 28,7\text{N}$$

$$\sum m_{cy} = 0 \Leftrightarrow -F_{ly1}.75 - F_{r2}.32 + F_{r1}.103 = 0 \Rightarrow F_{ly1} = -0,5\text{N}$$

Đường kính tại các tiết diện

Theo công thức (10.17):

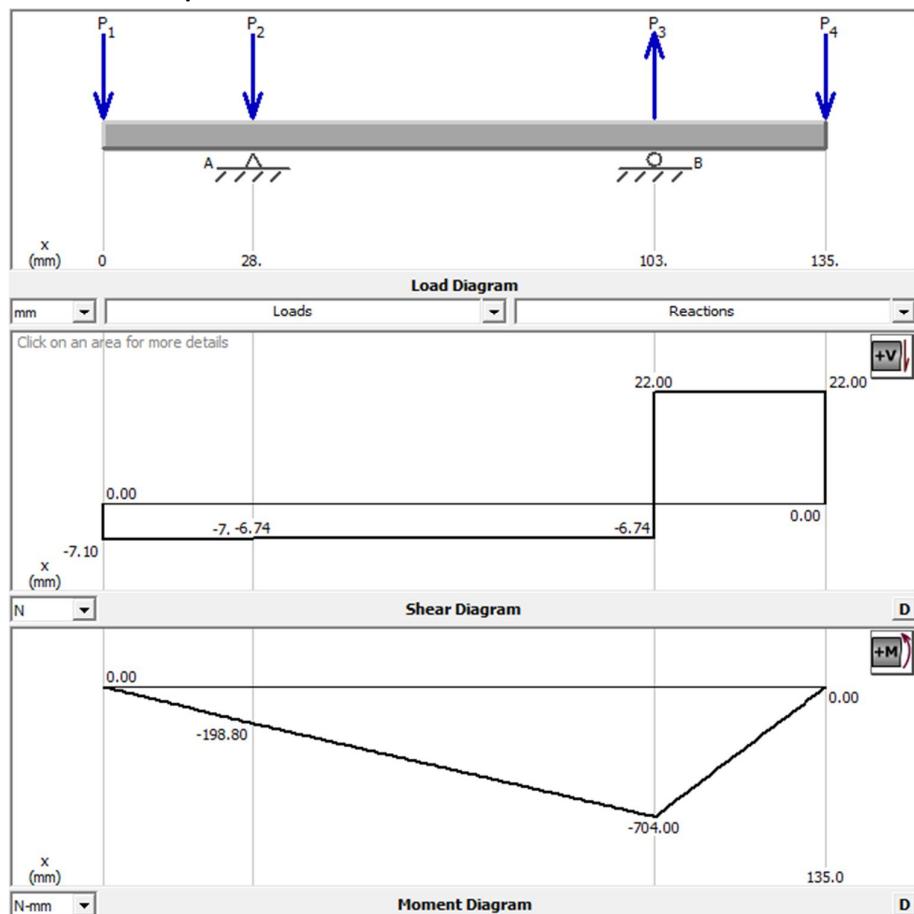
$$d_i \geq \sqrt[3]{\frac{\sqrt{M_{xi}^2 + M_{yi}^2 + 0,75T_i^2}}{0,1.[\sigma]}}$$

Trong đó, $[\sigma]$ - ứng suất cho phép của thép chế tạo trực cho trong bảng 10.5 tra được $[\sigma] = 20$ (CT3).

$$d_B \geq \sqrt[3]{\frac{\sqrt{198,8^2 + 0,75 \cdot 6844^2}}{0,1 \cdot 20}} = 14,36$$

$$d_C \geq \sqrt[3]{\frac{\sqrt{704^2 + 0,75 \cdot 6844^2}}{0,1 \cdot 58}} = 14,4$$

Chọn lại kích thước trục: $d = 15\text{mm}$



Hình 3.4.4.3 Momen lên trục

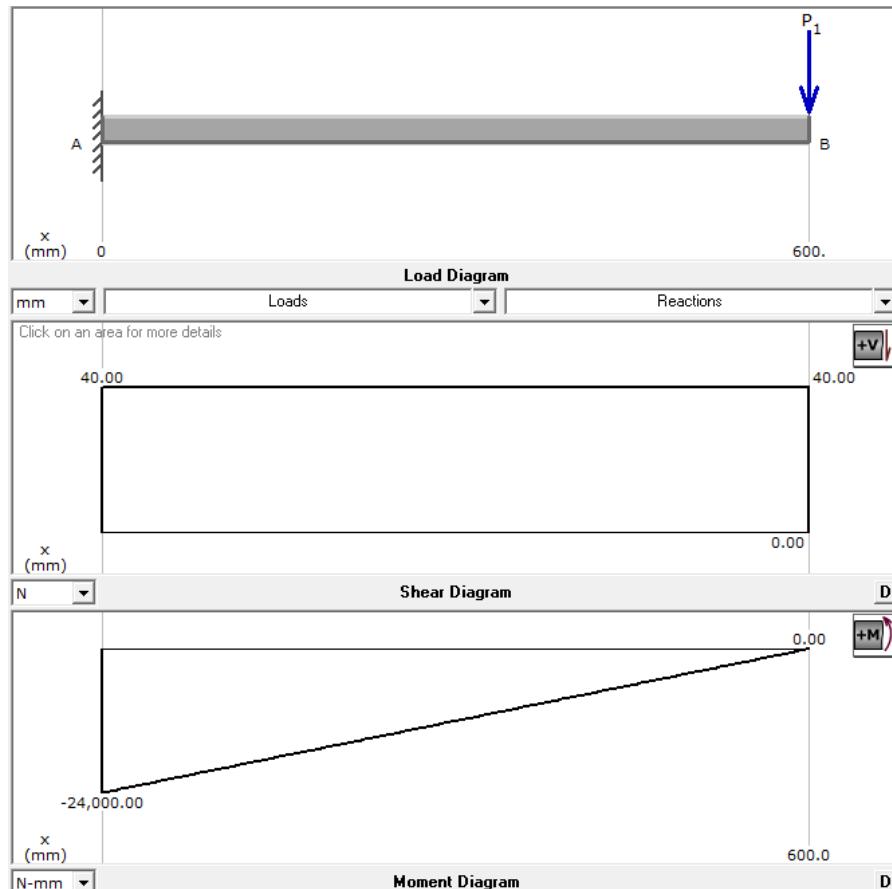
2.4.4) Chọn ô lăn

Dựa vào đường kính trục ta có chiều rộng ô lăn là 15mm
Và có lực dọc trục nên ta chọn ô lăn đỡ chặn

2.4.5) Xét tính bền của các thanh chịu lực chính

a) Thanh ngang chính của máy.

Tính toán các lực trên trục:



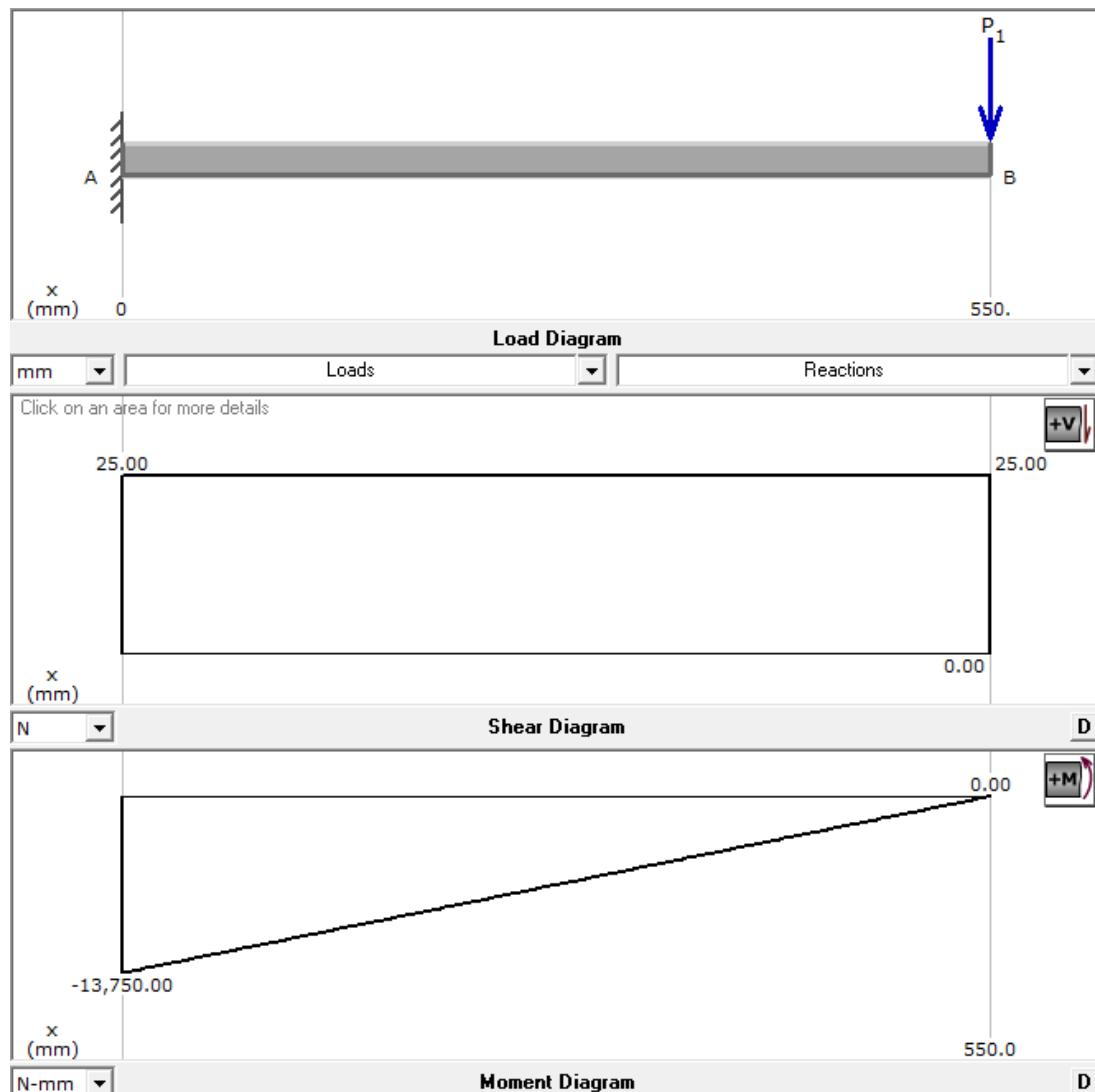
Hình 2.4.5a Momen lên thanh ngang chính của máy

Ta có Giới hạn chảy $\sigma_{0.2} \geq 210 \text{ N/mm}^2$

So sánh với momen lực do thanh chịu $\Rightarrow 2100.360 > 24000$

\Rightarrow Thanh đủ điều kiện bền

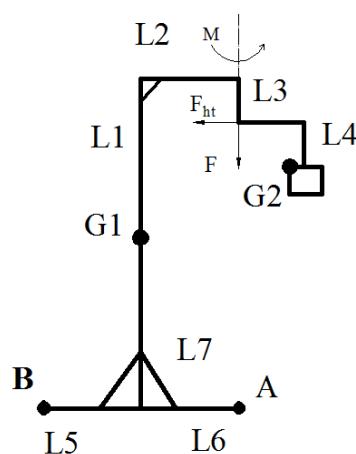
b) Thanh ngang cụm máy ảnh.



Hình 2.4.5b Momen lên thanh ngang cụm máy ảnh

Chứng minh tương tự phần a) ta có $2100.220 > 24000$
=> Thanh đủ điều kiện bền

2.4.6) Xem xét khả năng hệ bị lật



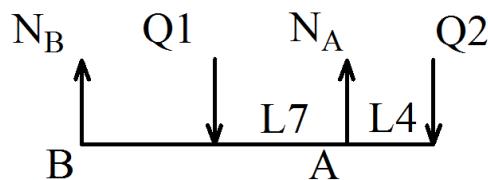
Ta xét hệ tại vị trí dẽ bị lật nhất

Với:

- G1: trọng tâm hệ thống thanh đỡ
- G2: trọng tâm hệ thống chuyển động.
- L1 = 1,2m
- L2 = 0,4m
- L3 = L4 = 0,3m
- L5 = L6 = 0,5m
- L7 = 0,3m

Hình 2.4.6a Xét khả năng vật lật

Biểu đồ lực:



Hình 2.4.6b Biểu đồ lực khi chiếu xuống mặt phẳng ngang

Vật sẽ lật quanh A khi: $N_B \leq 0$

Xét

$$\begin{aligned} \sum M_A = 0 &\Rightarrow Q2 \cdot L3 - Q1 \cdot L7 + N_B \cdot 2L7 = 0 \\ &\Rightarrow N_B = Q1 \cdot L7 - Q2 \cdot L3 / 2L7 \\ &\Rightarrow Q1 \cdot L7 - Q2 \cdot L3 \leq 0 \end{aligned}$$

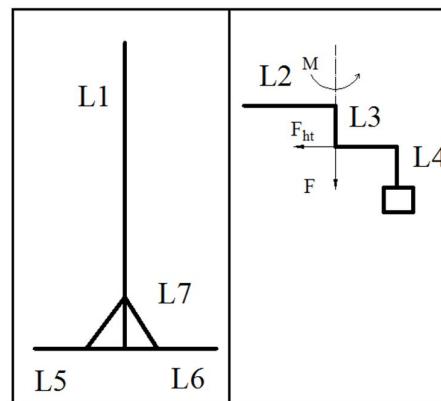
Mà: $Q1 = 69N$; $Q2 = 31N$

$69 \cdot 0,54 - 19 \cdot 0,4 > 0 \Rightarrow$ hệ vật không thể lật

Lực li tâm: $F = m\omega^2/R = 1,9 \cdot (2\pi/10)^2 / 0,9 = 0,6 \Rightarrow$ không đáng kể

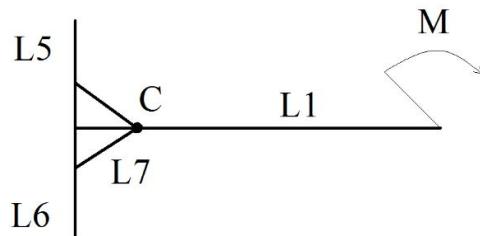
2.4.7) Tính toán độ bền của trụ đỡ chính

Tách hệ thành 2 phần như hình



Hình 2.4.7a Tách hệ vật

Sau khi tổng hợp các lực bên trái ta được momem tổng hợp ở đầu thanh



Hình 2.4.7b Momen tổng hợp

Với giá trị $M = 176 \text{ N.cm}$

Xét điều kiện bền theo ứng suất cho phép tại điểm C

$$|\sigma|_{max} = \frac{|M|_{max}}{W_x} \leq [\sigma]; W_x = 22,4$$

$$|\sigma|_{max} = \frac{176 \cdot 105}{22,4} = 825 \frac{\text{N}}{\text{cm}^2} \leq [\sigma] = 11 \text{ kN/cm}^2$$

\Rightarrow Trục đỡ chính đảm bảo điều kiện bền

CHƯƠNG 3: THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN VÀ ĐIỀU KHIỂN

Phần này sẽ đi vào phân tích và lựa chọn phương án điều khiển thích hợp, dựa trên một số yêu cầu về điều khiển ta sẽ lựa chọn các thiết bị phù hợp để có thể vận hành và giám sát tình trạng hoạt động của máy cắt cồng một cách hiệu quả.

3.1) Nguyên li hoạt động

3.1.1) Chế độ Auto

Khi bấm “Start” vi điều khiển Arduino Mega nhận tín hiệu điều khiển, sau đó xuất xung cấp cho mạch MSD_AH điều khiển động cơ Servo thực hiện chu trình đến từng vị trí đã đặt trước, khi đến điểm cuối cùng của chu trình, động cơ sẽ trả về lại vị trí ban đầu.

Khi bấm “Stop” động cơ đang ở vị trí ban đầu thì đứng yên, còn đang ở vị trí bắt kè thì động cơ tự động trả về điểm ban đầu

3.1.2) Chế độ Normal

Khi bấm “0°”, “90°”, “180°” và “270°” mạch driver MSD_AH sẽ nhận xung điều khiển từ vi điều khiển Arduino Mega điều khiển động cơ đến lần lượt đến các vị trí ban đầu, 90°, 180° và 270°

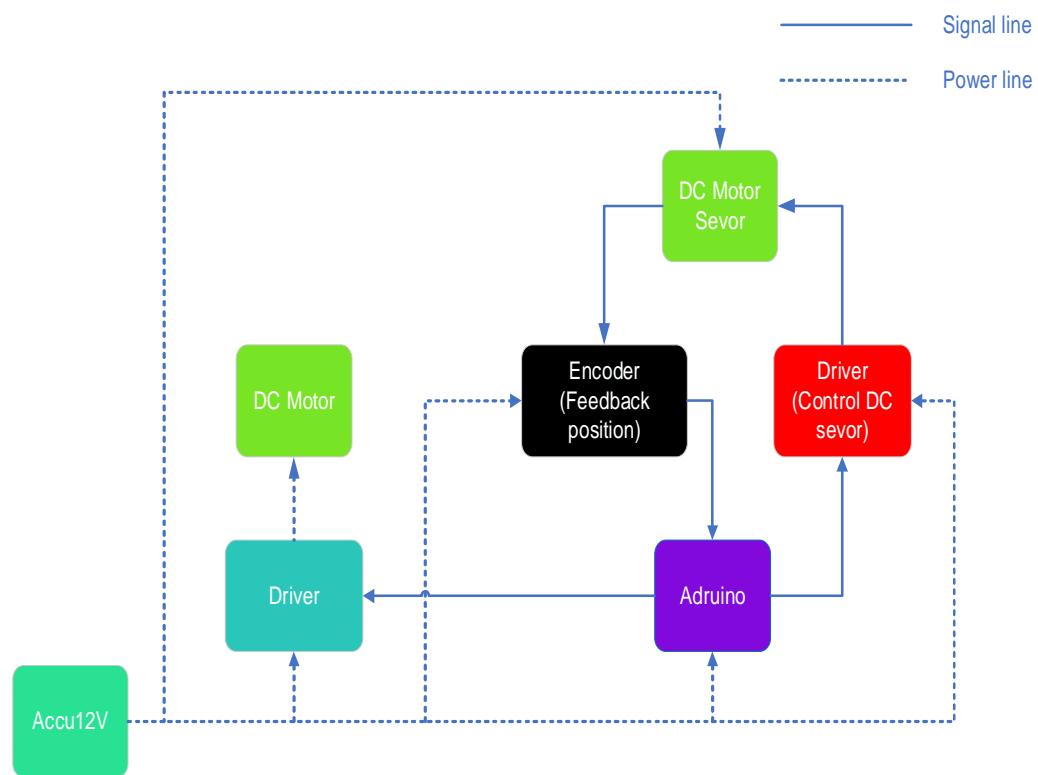
3.1.3) Nâng cơ cầu chụp

Khi bấm “up” mạch điều khiển động cơ DC BTS7960 nhận tín hiệu để điều khiển động cơ nâng hệ thống lên

Khi bấm “down” hệ thống sẽ được hạ xuống do tín hiệu cấp cho BTS 7960 từ vi điều khiển

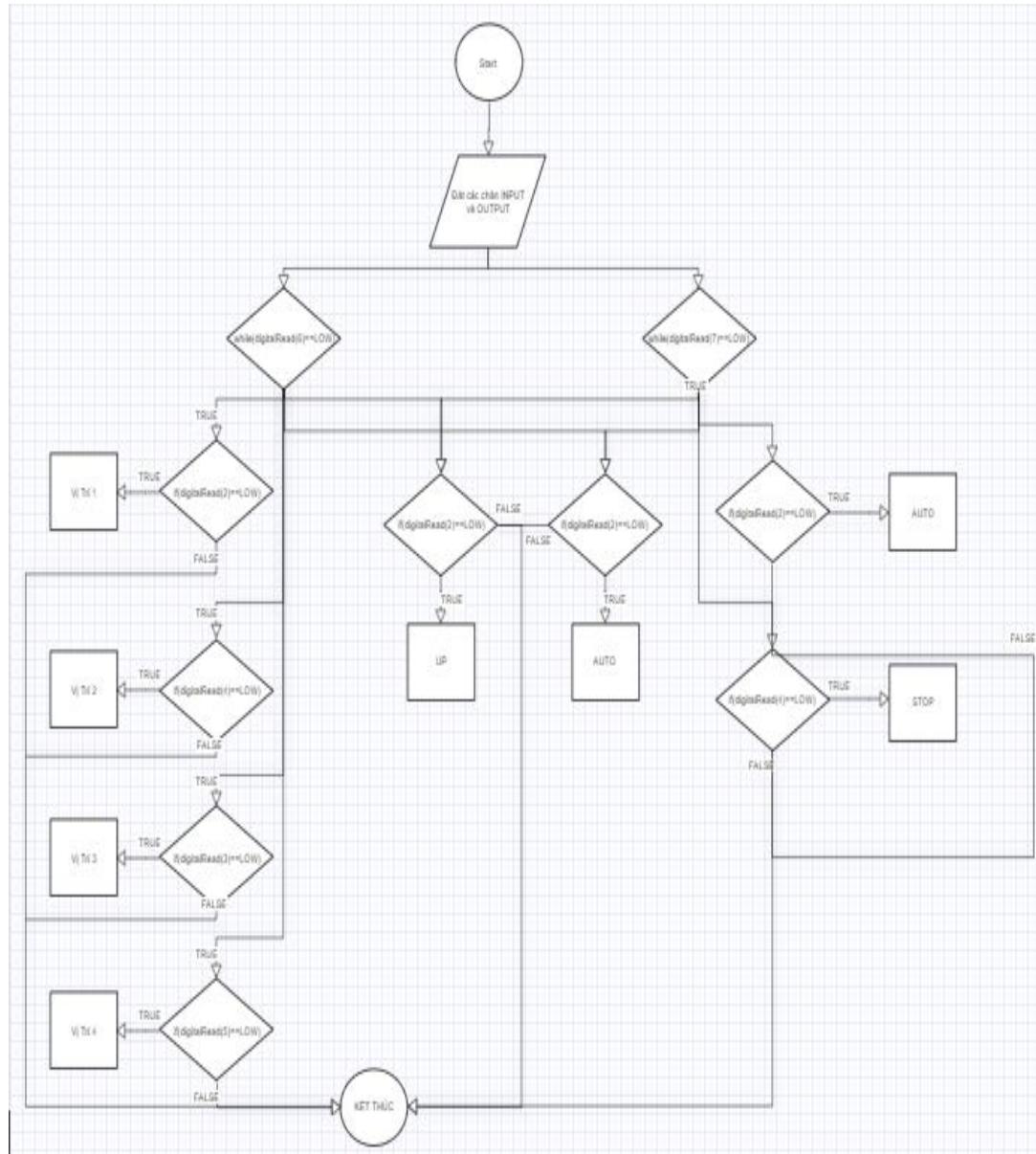
3.2) Cơ cấu bộ điều khiển

3.2.1) Sơ đồ khái niệm điều khiển



Hình 3.2.1 Sơ đồ điều khiển

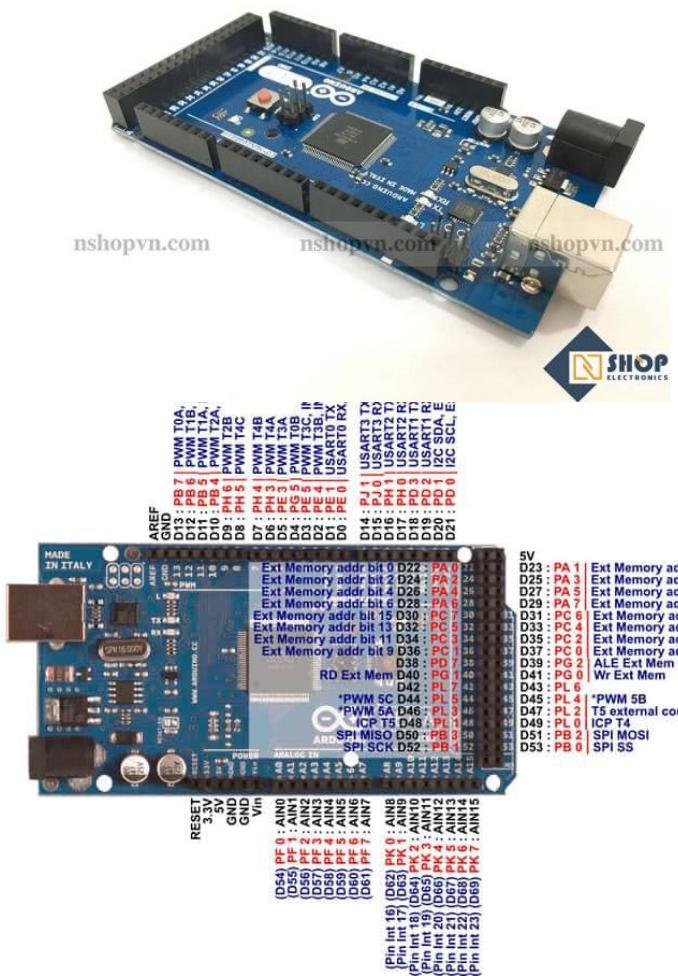
3.2.2) Sơ đồ giải thuật



Hình 3.2.2 Sơ đồ giải thuật

3.3) Lựa chọn các phương án điều khiển

3.3.1) Vi xử lí Adruino Mega



Hình 3.3.1 Vi xử lí Adruino Mega

3.3.1.1) Giới thiệu về vi xử lí Adruino

Mạch điện Arduino Mega 2560 R3 Atmega16u2 là phiên bản nâng cấp của Arduino Uno R3 với số chân giao tiếp, ngoại vi và bộ nhớ nhiều hơn, mạch được thiết kế và sử dụng các linh kiện tương đương với phiên bản chính hãng trên Arduino.cc, phù hợp cho các ứng dụng cần nhiều bộ nhớ hoặc nhiều chân, cổng giao tiếp hơn so với Arduino Uno

Mạch Arduino Mega 2560 Atmega là một board vi điều khiển dựa trên ATmega2560. Board này có 54 chân I/O (14 chân PWM), 16 analog đầu hàng vào, 4 UARTs (phản ứng công suất tuần tự), sử dụng thạch anh 16 MHz, kết nối cổng USB,

một Jack cắm điện, chân ICSP, và một nút reset. Board có tất cả mọi thứ cần thiết để hỗ trợ vi điều khiển.

3.3.1.2) Thông Số Kỹ Thuật

Ví điều khiển chính: ATmega2560

IC nạp và giao tiếp UART: ATmega16U2.

Nguồn nuôi mạch: 5VDC từ cổng USB hoặc nguồn ngoài cắm từ giắc tròn DC (khuyên dùng 7-9VDC để đảm bảo mạch hoạt động tốt. Nếu bạn cắm 12V thì IC ổn áp rất dễ chết và gây hư hỏng mạch).

Số chân Digital: 54 (15 chân PWM)

Số chân Analog: 16

Giao tiếp UART : 4 bộ UART

Giao tiếp SPI : 1 bộ (chân 50 -> 53) dùng với thư viện SPI của Arduino

Giao tiếp I2C : 1 bộ

Ngắt ngoài : 6 chân

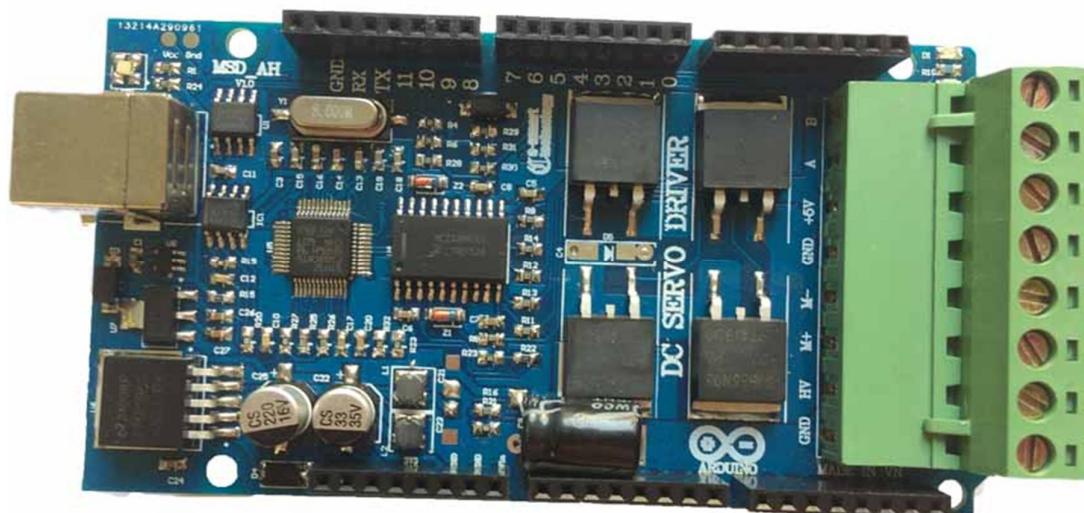
Bộ nhớ Flash: 256 KB, 8KB sử dụng cho Bootloader

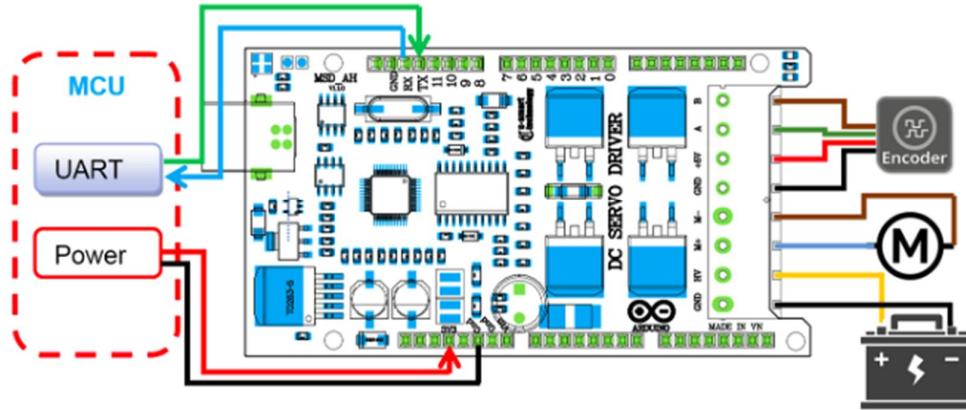
SRAM: 8 KB

EEPROM: 4 KB

Xung clock: 16 MHz

3.3.2) Driver điều khiển động cơ Servo





Hình 3.3.2 Driver điều khiển động cơ Servo

3.3.2.1) Giới thiệu về Driver MSD_AH

MSD_AH là module điều khiển động cơ DC Servo, được thiết kế chuyên biệt hỗ trợ tương thích với Arduino. Mỗi một driver MSD_AH sẽ được đánh một địa chỉ cố định nhờ đó người sử dụng có thể dễ dàng điều khiển cùng lúc nhiều động cơ khác nhau với một board chủ Arduino thông qua mạng UART.

Với sự hỗ trợ hết sức mạnh mẽ của nhân ARM Coretex-M3, công nghệ Mosfet tiên tiến, đặc biệt sự kết hợp các thuật toán kinh điển về điều khiển vòng kín. Chúng tôi cho ra đời sản phẩm điều khiển động cơ DC Servo với độ chính xác cao. Sản phẩm tích hợp nhiều thuật toán điều khiển vòng kín và khôi công suất trên cùng một board mạch. Giúp board mạch hết sức nhỏ gọn và giảm chi phí đáng kể cho người sử dụng.

Với một driver bạn có thể sử dụng để điều khiển vận tốc hoặc vị trí và vận tốc.

Đặc biệt driver được kèm theo một phần mềm, tích hợp công cụ tự động dò tìm thông số của hệ thống từ đó đưa ra các chuẩn đoán về các hệ số điều khiển của động cơ. Phần mềm còn hỗ trợ người dùng quan sát, kiểm thử đáp ứng của hệ thống một cách trực quan và trực tiếp.

3.2.2.1) Thông số kỹ thuật

Tương thích với Arduino Uno, Arduino Mega.

Phương pháp điều khiển Master (Arduino) to many slave (MSD_AH) 10-30VDC, 0-20A, 0-400W

Điều khiển vị trí, vận tốc, gia tốc

Encoder x4, tích hợp bộ lọc nhiễu encoder

Phương thức giao tiếp: UART, USB

Tích hợp các bộ điều khiển vòng kín: PID, PI, State feedback

Tích hợp công cụ tự động giờ tìm thông số PID cho động cơ

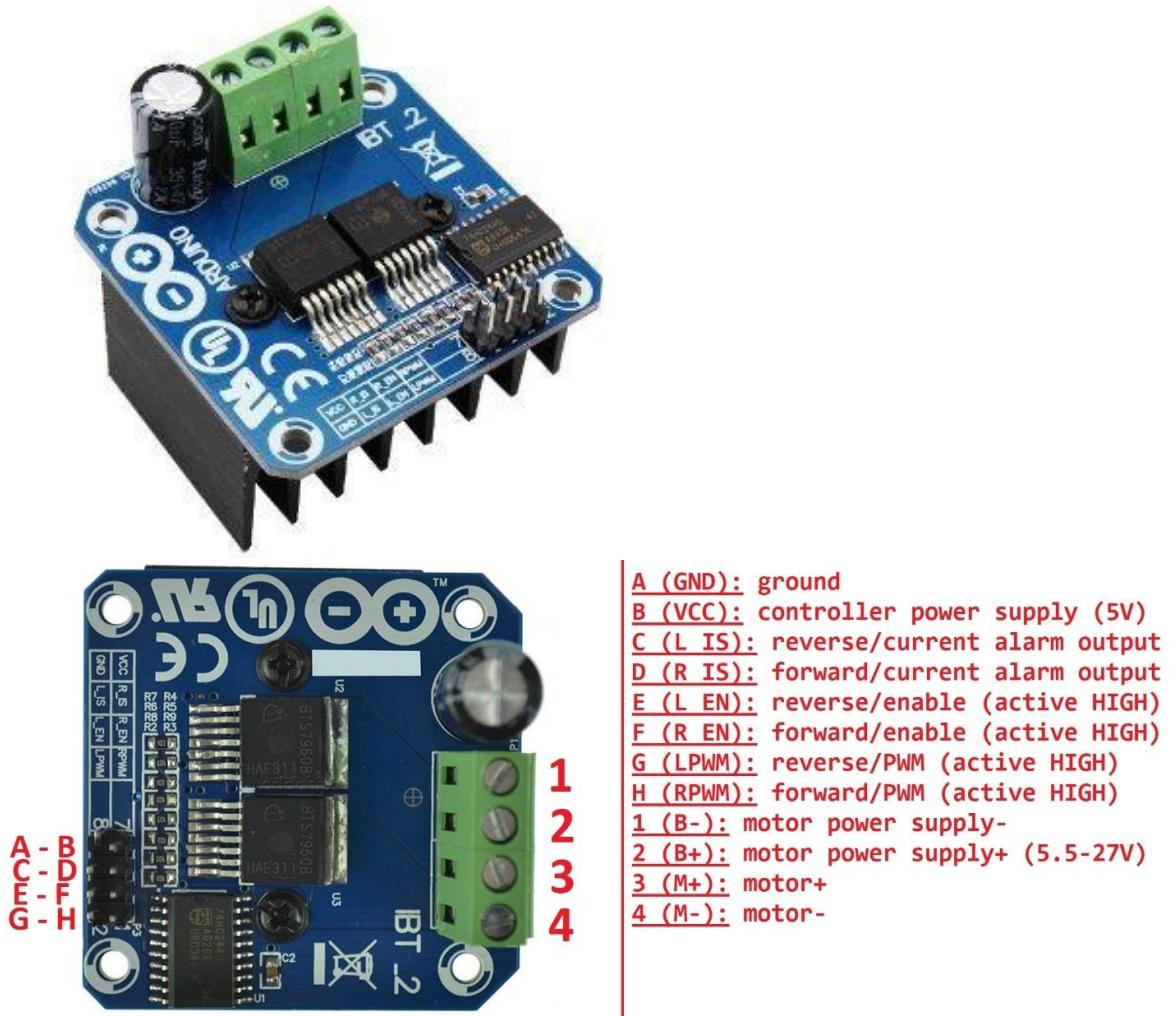
Tích hợp nhiều chế độ bảo vệ: dòng điện, trễ pha...

3.3.2.3) Lệnh điều khiển

BẢNG 3.3.2.3 LỆNH ĐIỀU KHIỂN

Các thông số	Ý nghĩa
Nx	x là địa chỉ của driver (x=0 là địa chỉ Broadcast)
Py	y là vị trí muốn điều khiển đến (rad)
Vz	z là vận tốc (rad/s)
Aj	j là gia tốc (rad/s ²)
G	Lấy vị hiện tại, driver sẽ truyền UART giá trị của vị trí.
S	Dừng
r	Reset vị trí
Mx	Thay đổi mode hoạt động, x = mode
Tx	Thay đổi duty cycle

3.3.3) Modul điều khiển động cơ DC BTS 7960



Hình 3.3.3 Modul điều khiển động cơ DC BTS 7960

3.3.3.1) Giới thiệu về Modul BTS7960

Mạch điều khiển động cơ DC BTS7960 43A dễ dàng giao tiếp với vi điều khiển với driver tích hợp sẵn trong IC với đầy đủ các tính năng current sense (kết hợp với điện trở đo dòng), tạo dead time, chống quá nhiệt, quá áp, quá dòng, sụt áp và ngắn mạch.

3.3.3.2) Thông số kỹ thuật

Điện áp đầu vào : 6~27V

Dòng điện tải mạch: 43A

Tín hiệu logic điều khiển: 3.3 ~ 5V.

Tần số điều khiển tối đa: 25KHz.

Tự động shutdown khi điện áp thấp: để tránh điều khiển động cơ ở mức điện áp thấp thiết bị sẽ tự shutdown. Nếu điện áp < 5.5V, mạch điều khiển động cơ DC BTS7960

sẽ tự ngắt điện và sẽ mở lại sau khi điện áp > 5.5V.

Bảo vệ quá nhiệt: BTS7960 bảo vệ chống quá nhiệt bằng cảm biến nhiệt tích hợp bên trong. Đầu ra sẽ bị ngắt khi có hiện tượng quá nhiệt.

Kích thước: 40 x 50 x12mm.

3.3.3.3) Sơ đồ chân

VCC : Nguồn tạo mức logic điều khiển (5V – 3V3)

GND : Chân đất.

R_EN = 0 Disable nửa cầu H phải. R_EN = 1 : Enable nửa cầu H phải.

L_EN = 0 Disable nửa cầu H trái. L_EN = 1 : Enable nửa cầu H trái.

RPWM và LPWM : chân điều khiển đảo chiều và tốc độ động cơ.

RPWM = 1 và LPWM = 0 : Mô tơ quay thuận.

RPWM = 0 và LPWM = 1 : Mô tơ quay nghịch

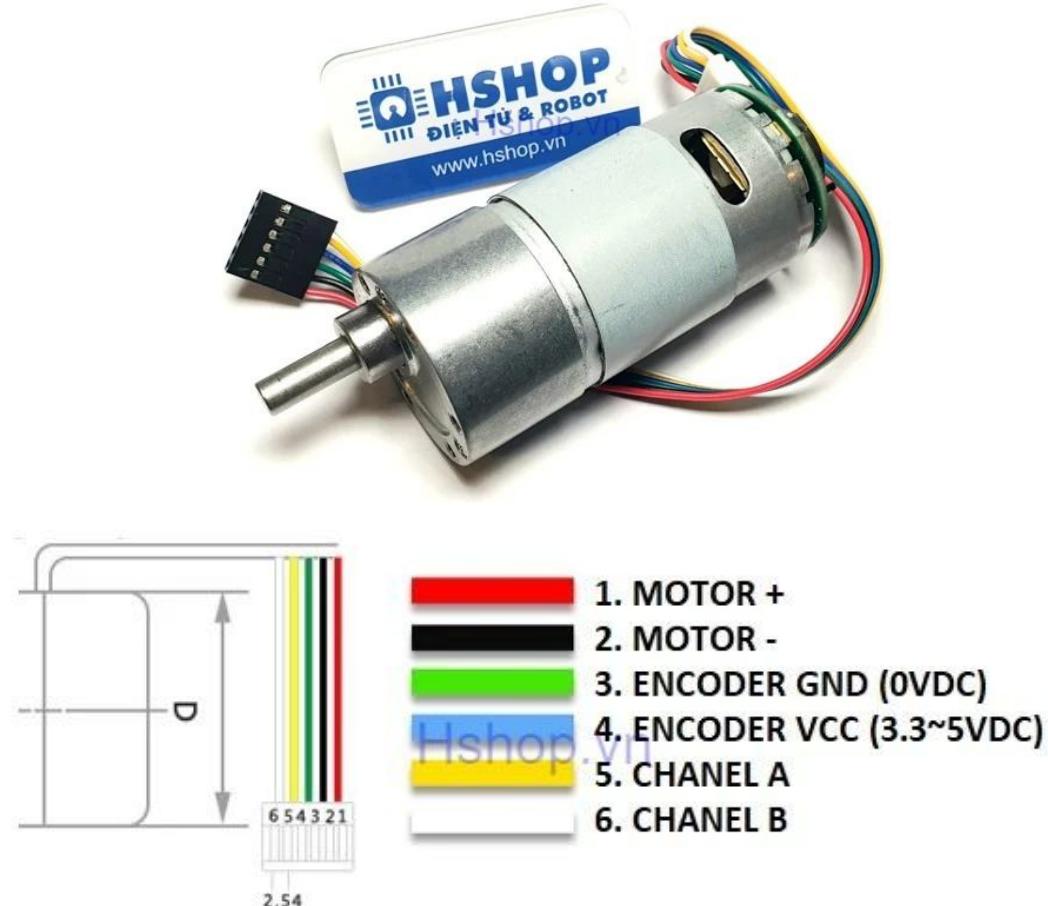
RPWM = 1 và LPWM = 1 hoặc RPWM = 0 và LPWM = 0 : Dừng.

R_IS và L_IS : kết hợp với điện trở để giới hạn dòng qua cầu H

Với ứng dụng bình thường RPWM, LPWM nối với GPIO(vd: chân digital 2,3) để điều khiển chiều quay của động cơ

Chân R_EN , L_EN nối chung lại rồi nối với PWM (VD chân digital 5) để điều khiển tốc độ động cơ.

3.3.4) Động cơ DC Servo



Hình 4.3.4 Động cơ DC Servo

3.3.4.1) Giới thiệu về Động cơ DC Servo

Động cơ DC Servo JGB37-545 DC Geared Motor được tích hợp thêm Encoder hai kênh AB giúp đọc và điều khiển chính xác vị trí, chiều quay của động cơ trong các ứng dụng cân độ có chính xác cao: điều khiển PID, Robot tự hành,....

Động cơ DC Servo JGB37-545 DC Geared Motor có cấu tạo bằng kim loại cho độ bền và độ ổn định cao, được sử dụng trong các mô hình robot, xe, thuyền,..., hộp giảm tốc của động cơ có nhiều tỉ số truyền giúp bạn dễ dàng lựa chọn giữa lực kéo và tốc độ (lực kéo càng lớn thì tốc độ càng chậm và ngược lại), động cơ sử dụng nguyên liệu chất lượng cao (lõi dây đồng nguyên chất, lá thép 407, nam châm từ tính mạnh,...) cho sức mạnh và độ bền vượt trội hơn các loại giá rẻ trên thị trường hiện nay (sử dụng lõi dây nhôm, nam châm từ tính yếu).

3.3.4.2) Thông số kỹ thuật:

Điện áp sử dụng: 12VDC

Đường kính: 37mm

Encoder: Cảm biến từ trường Hall, có 2 kênh AB lệch nhau giúp xác định chiều quay và vận tốc của động cơ, mỗi Encoder trả ra 11 xung/1 kênh/ 1 vòng (nếu đo tín hiệu đồng thời của cả hai kênh sẽ thu được tổng 22 xung / 1 vòng quay của Encoder).

Cách tính số xung của mỗi kênh trên 1 vòng quay của trục chính động cơ = Tỉ số truyền x số xung của Encoder, ví dụ tỉ số 150:1 thì số xung Encoder trả ra cho 1 vòng quay của trục chính động cơ sẽ là $11 \times 150 = 1650$ xung / 1 kênh.

Điện áp cấp cho Encoder hoạt động: 3.3~5VDC, lưu ý cấp quá áp hoặc ngược chiều sẽ làm cháy Encoder ngay lập tức!

Loại 12VDC 37RPM:

Tỉ số truyền 168:1 (động cơ quay 168 vòng trục chính hộp giảm tốc quay 1 vòng).

Dòng không tải: 200mA

Dòng chịu đựng tối đa khi có tải: 5A

Tốc độ không tải: 37RPM (37 vòng 1 phút)

Tốc độ chịu đựng tối đa khi có tải: 30RPM (30 vòng 1 phút)

Lực kéo Moment định mức: 21KG.CM

Lực leo Moment tối đa: 84KG.CM

Chiều dài hộp số L: 26.5mm

Số xung Encoder mỗi kênh trên 1 vòng quay trục chính: $11 \times 168 = 1848$ xung.

3.4.5) Động cơ DC 12/24V



Hình 3.4.5 Động cơ DC Servo

3.4.5.1 Giới thiệu về động cơ DC

Động cơ điện một chiều này Ô đĩa siêu mượt, gần như không có tiếng ồn, an toàn sử dụng. Với tốc độ cao 3500/7000rpm cho sự lựa chọn của bạn và mô-men xoắn lớn. Được sử dụng rộng rãi trong máy làm kẹo bông, bàn cắt nhỏ, máy mài và các thiết bị cơ khí khác.

Các tính năng:

Nó lái siêu mượt, hầu như không có tiếng ồn, an toàn khi sử dụng.

Với tốc độ cao 3500/7000rpm cho sự lựa chọn của bạn và mô-men xoắn lớn.

Cuộn dây Stator của động cơ là dây đồng, Nhiệt độ thấp, tổn thất thấp, tuổi thọ cao, Và nó có thể nhận ra CW/CCW bằng điều khiển công tắc (không bao gồm công tắc).

Cần cho Bộ điều chỉnh dòng điện trực tiếp bên ngoài, nó có thể đạt được chức năng điều chỉnh tốc độ.

Được sử dụng rộng rãi trong máy làm kẹo bông, bàn cắt nhỏ, máy mài và các thiết bị cơ khí khác.

Vòng bi, ma sát nhỏ, vòng đệm dầu mạnh, khả năng chịu lực cao

3.4.5.2) Thông số kỹ thuật:

Loại sản phẩm: động cơ DC chổi than

Mẫu: XD-3420

Công suất định mức: 30W

Điện áp định mức (Tùy chọn): 12V, 24 V

Tốc độ định mức: 3500Rpm (12V), 7000 RPM(24V)

Hiện tại: 0.5 A

Chiều dài trục đầu ra: 26mm/1.02inch

Mô-men xoắn: 1kgf.cm(12V), 2kgf.cm(24V)

Trọng lượng: 460 gam

3.4.6) Cầu dao tự động

Thuộc dòng cầu dao tự động MCB, tự động bảo vệ quá tải và ngắn mạch 6kA.



Hình 3.4.6 Cầu dao tự động

Sử dụng phổ thông trong các mạch đóng ngắt điện tổng dân dụng hoặc công nghiệp.

Tự động ngắt điện khi dòng phụ tải vượt quy định

3.4.7) Nút nhấn



Hình 3.4.7 Nút nhấn

CR Series nút nhấn chuyên mạch sử dụng trong các mạch điều khiển công nghiệp của AC 50Hz hoặc 60Hz, dài điện áp hoạt động 380V, DC điện áp hoạt động 220V và điện áp thấp hơn để điều khiển cho khởi động điện từ, contactor, relay và mạch điện khác.

Các Nút nhấn có đèn báo có thể được sử dụng trong các lĩnh vực, nơi được chỉ thị bởi tín hiệu ánh sáng

Dài Điện áp: AC 50Hz/60Hz, AC380V/DC220V

-Tiêu chuẩn: IP40

Tiêu chuẩn: IEC/EN60947-5-1

Thông số kỹ thuật Nút nhấn CR-251-1(CRF-F25M1)

Nút nhấn mặt nhôm

Lỗ khoét: Phi 25

Số tiếp điểm: 1NO + 1NC (1a1b)

Dòng điện tiếp điểm: 5A

Điện áp hoạt động: AC/DC 230, 380

Điện trở tiếp xúc: $\leq 50m\Omega$

Tuổi thọ đóng ngắt điện: 1×10^5

Tuổi thọ đóng ngắt cơ khí: 1×10^5

Nhiệt độ làm việc: $-5^\circ C \div 50^\circ C$

Độ ẩm làm việc: 45% \div 90%

3.4.8) Nguồn

Bộ nguồn tổ ong 12V 5A chuyển đổi điện AC 220V sang điện áp DC 12V thông

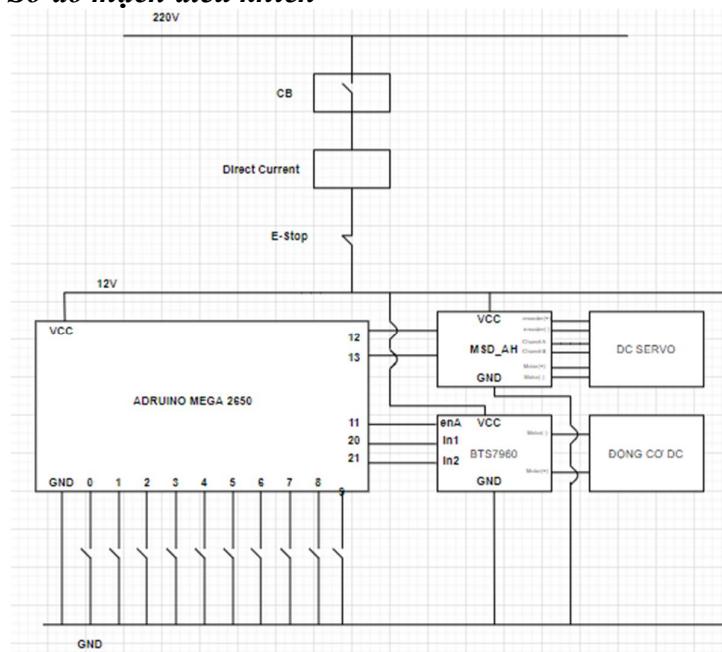


Hình 3.4.8 Nguồn

dụng cho nhiều thiết bị. Nguồn tổ ong 12V 5A với thiết kế nhỏ gọn, có chiết áp thường được dùng cho các mạch công suất, đèn led, tủ điều khiển tự động hoặc camera,....

3.5) Thiết kế mạch điện cho hệ thống

3.5.1) Sơ đồ mạch điều khiển

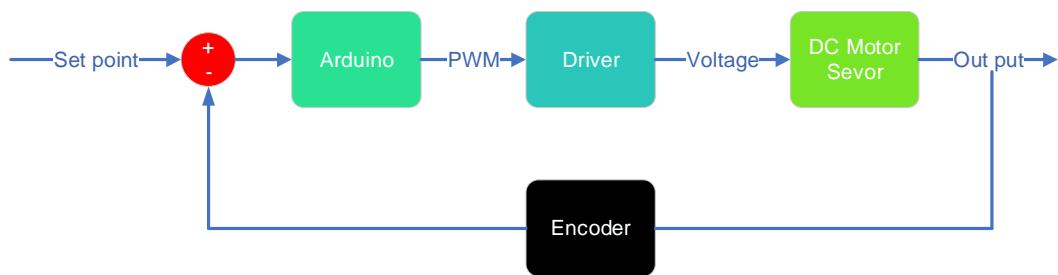


Hình 3.5.1 Sơ đồ mạch điều khiển.

3.5.2) Lưu đồ điều khiển vị trí

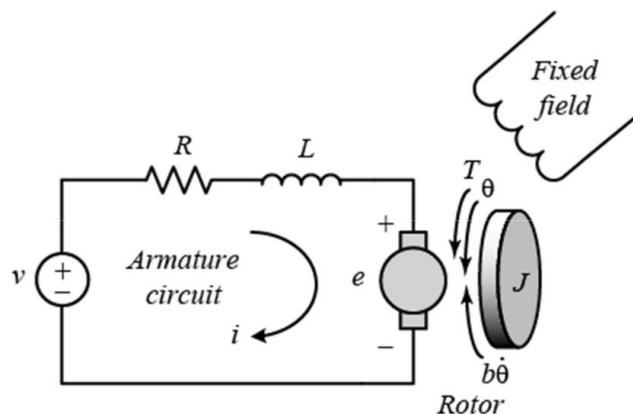
Khi điều khiển một hệ thống bất kỳ, tham số cần quan tâm đầu tiên đó là vị trí và tốc độ, điều khiển tốc độ và vị trí để đảm bảo tốc độ và vị trí thực của động cơ luôn đáp ứng được yêu cầu cho trước, trong bài toán này vị trí là đại lượng chính cần sét đến vì nó đóng vai trò quan trọng trong việc điều khiển vị trí đúng với yêu cầu của người sử dụng. Dưới đây là lưu đồ điều khiển vị trí bằng driver MSD_AH thông qua động cơ DC

Sơ đồ điều khiển



Hình 3.5.2 Lưu đồ điều khiển vị trí

3.5.3) Tính toán thiết kế bộ điều khiển PID



Hình 3.5.3a Tính toán thiết kế bộ điều khiển PID

Ta có :

Tín hiệu đầu vào của hệ thống là Volt (V).

Đơn vị đầu ra vận tốc góc CƠ của động cơ (rad/s).

Phương trình mạch điện phần cứng

$$U = L \frac{di}{dt} + Ri + E_{ew}$$

Trong đó:

R là điện trở phần ứng (Ω).

L là điện cảm phần ứng (H).

i là dòng điện phần ứng (A).

E_{ew} Sức điện động ở phần cứng (V)

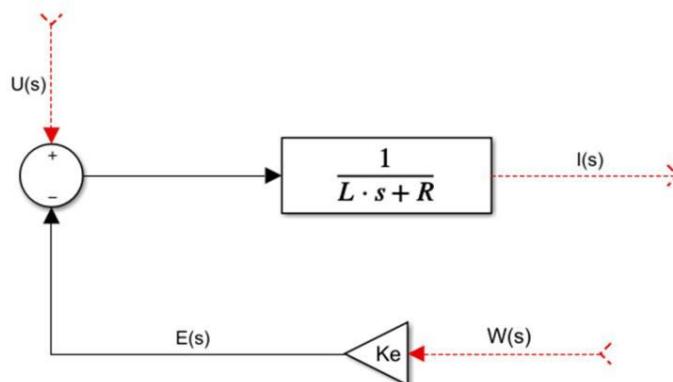
Biến đổi Laplace hai vế của phương trình, ta được:

$$U(s) = L(s) + RI(s) + E(s)$$

$$\Leftrightarrow U(s) - E(s) = (Ls + R)I(s)$$

$$\Leftrightarrow I(s) = \frac{U(s) - E(s)}{Ls + R}$$

Phương trình Moment điện động cơ



Hình 3.5.3b Tính toán thiết kế bộ điều khiển PID

Moment điện từ của động cơ tỉ lệ với dòng điện phần ứng:

$$M = K_m i$$

Trong đó:

K_m là hằng số moment của động cơ được tính bằng công thức

$$K_m = k_1 k_2 k_3 = k_1 \phi \text{ (N.m/A)}$$

(k_1 là hằng số phụ thuộc vào kết cấu động cơ, k_2 là hằng số đặc trưng tuyến tính của từ thông thay đổi theo i_k).

Biến đổi Laplace hai vế của phương trình trên ta được:

$$M(s) = K_m I(s)$$

Ta lập sơ đồ khói tương ứng:



Phương trình cân bằng moment trên trực động cơ:

$$M = J \frac{d\omega}{dt} + B\omega + M_t$$

Trong đó:

J là momen quán tính của động cơ và tải quy về trực động cơ ($Kg \cdot m^2$)

B là hệ số ma sát nhớt của động cơ và tải quy về trực động cơ ($Kg \cdot m^2$)

M_t là moment nhiễu ($N.m$)

Biến đổi Laplace hai vè ta được:

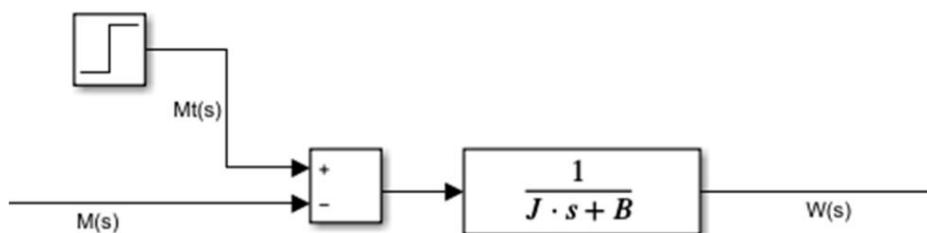
$$M(s) = Js\omega(s) + B\omega(s) + M_t(s)$$

$$\Leftrightarrow M(s) - M_t(s) = (Js + B)\omega(s)$$

$$\Leftrightarrow \omega(s) = \frac{M(s) - M_t(s)}{Js + B}$$

Sơ đồ khói tương ứng:

Hàm truyền của động cơ DC với tín hiệu ra là vận tốc góc:



$$G(s) = \frac{\omega(s)}{U(s)} = \frac{\frac{K_m}{(Ls + R)}}{1 + \frac{K_m \cdot K_e}{(Ls + R)(Js + B)}} = \frac{K_m}{(Ls + R)(Js + B) + K_m \cdot K_e}$$

Hay:

$$G(s) = \frac{\omega(s)}{U(s)} = \frac{K_m}{LJ + (LB + RJ)s + K_m \cdot K_e + RB}$$

Đặt

$\tau_t = \frac{L}{R}$ là hằng số thời gian điện từ

$\tau_c = \frac{J}{B}$ là hằng số thời gian cơ

$$\begin{aligned} G(s) &= \frac{\omega(s)}{U(s)} \frac{K_m}{RB(\tau_t + 1)(\tau_c + 1) + K_m \cdot K_e} \\ \Leftrightarrow G(s) &= \frac{\frac{K_m}{RB}}{\tau_t \tau_c s^2 + (\tau_t + \tau_c)s + (1 + \frac{K_m \cdot K_e}{RB})} \end{aligned}$$

Hay:

$$G(s) = \frac{\omega(s)}{U(s)} = \frac{K_{td}}{T_1 s^2 + T_2 s + 1}$$

Trong đó

$$T_1 = \frac{\tau_t \tau_c RB}{K_m \cdot K_e + RB'}$$

$$T_2 = \frac{(\tau_t + \tau_c)RB}{K_m \cdot K_e + RB'}$$

$$T_3 = \frac{K_m}{K_m \cdot K_e + RB'}$$

Tín hiệu đầu vào:

Vận tốc góc $\omega(t)$ (rad/s)

Tín hiệu đầu ra:

Vị trí $y(t)$ của máy ảnh

Gọi pz (m) là vị trí của máy ảnh, ta có phương trình quan hệ:

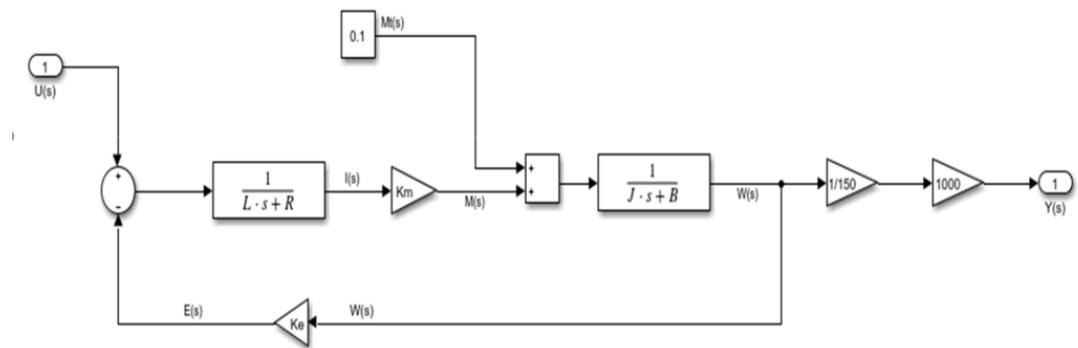
$$y(t) = \frac{P}{2\pi} \int_0^t \omega(t) dt$$

Biến đổi Laplace hai vế với điều kiện đầu bằng 0, ta được:

$$\frac{Y(s)}{\omega(s)} = \frac{P}{2\pi} \cdot \frac{\omega(s)}{s}$$

Sơ đồ khói miêu tả toàn bộ hệ thống:

Ta kết nối các sơ đồ khói của các phần tử lại và được sơ đồ khói của hệ thống như sau:



Hình 3.5.3c Tính toán thiết kế bộ điều khiển PID

3.5.4) Tính toán PID

Sử dụng phương pháp Zeigler-Nichols:

Cho một hàm nắc $r(t) = 10$ vào hệ thông động cơ DC ta được hệ số:

$$K = K_{gh} = 167$$

$$T_{gh} = 61,03678$$

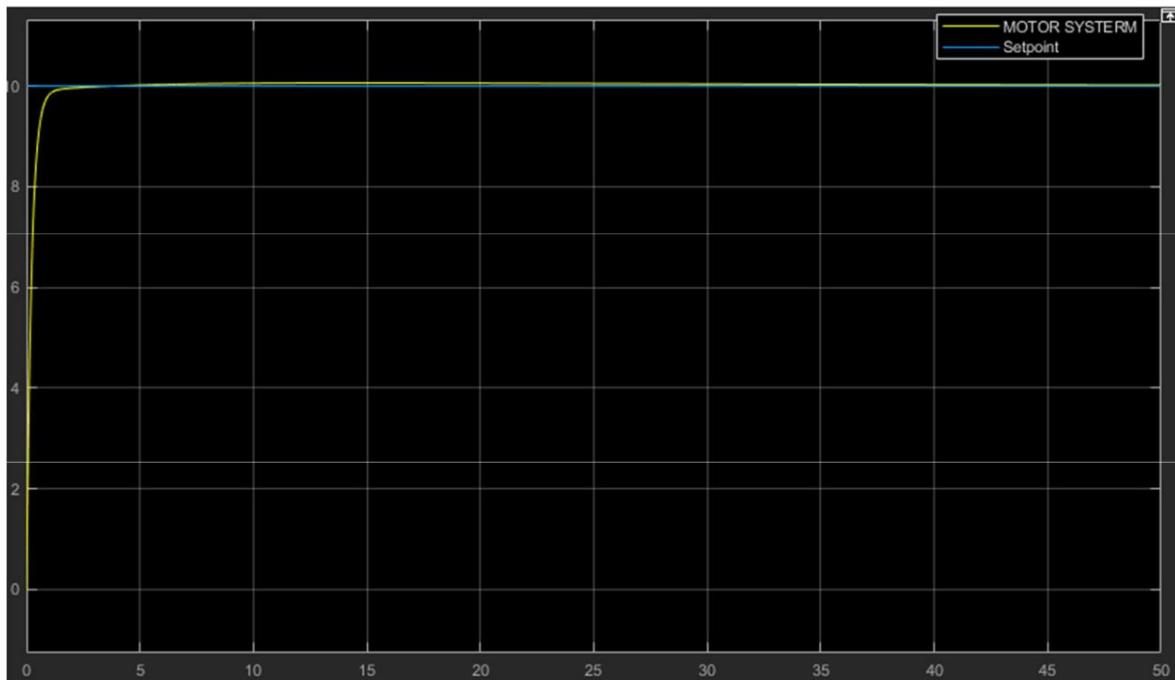
→ Từ 2 thông số trên ta suy ra được: $K_p = 0,6K_{gh}$, $T_i = 0,5T_{gh}$, $T_d = 0,125T_{gh}$
 $\Rightarrow K_p = 100,2 T_i$, $T_i = 30,64$, $T_d = 7,66$

Ta có:

$$\frac{U(s)}{E(s)} = K_p + K_i \frac{1}{s} + K_d \cdot s$$

$$= K_p + \frac{K_p}{T_i} \cdot \frac{1}{s} + K_p K_d \cdot s$$

$$\Rightarrow K_i = \frac{K_p}{T_i} = \frac{100,2}{30,64}, K_d = K_p T_d = 100,2 \cdot 4,41 = 411,82$$



Hình 3.5.4 Biểu đồ PID

3.5.6 Lập trình cho hệ thống trên DC TunerPro.

Chương trình thực tế:

Ta sử dụng mạch MSD_AH để đọc các giá trị đầu vào của sensor và xuất các giá trị đầu ra mong muốn. Dưới đây là sơ đồ khói của hệ chạy trong thời gian thực



(Kết quả thực nghiệm chạy thuận chạy nghịch PID)

CHƯƠNG 4: CƠ SỞ LÍ THUYẾT VÀ THUẬT TOÁN NHÂN DẠNG NHÂN TRẮC

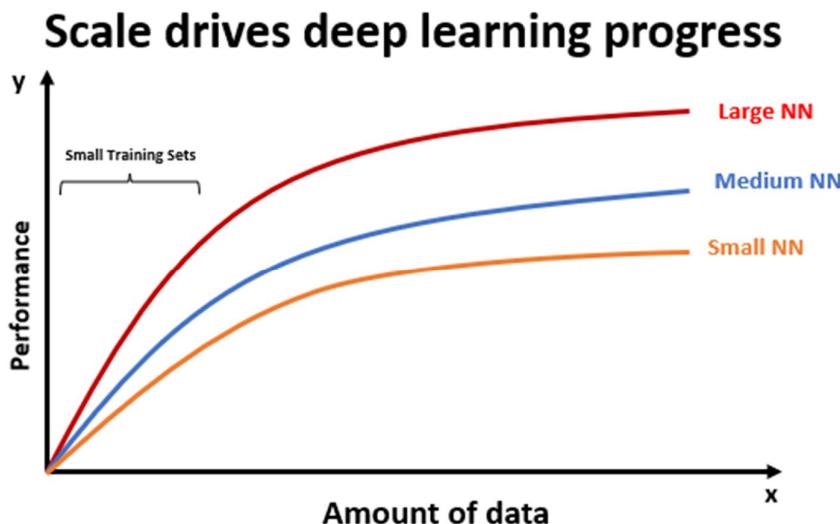
Chương này đưa ra lý thuyết liên quan, nguyên lý hoạt động của các thuật toán từ đó đưa ra mô hình hoạt động của YOLOv4 trong phân loại các vùng nhân trắc. Phần 4.1 và 4.2 sẽ giới thiệu cơ bản về Deep Learning, Neural network và thuật toán Convolution Neural Network, phần 4.3 sẽ giới thiệu về thuật toán YOLOv4 một mô hình CNN để phát hiện các đối tượng một cách nhanh chóng và hiệu quả hơn.

4.1) Khái niệm thuật toán liên quan.

4.1.1) Deep Learning và Machine Learning.

a) Deep Learning

Deep Learning (Học sâu) là một nhánh nhỏ của Machine Learning (Máy học), bắt nguồn từ thuật toán Neural Network. Tuy nhiên, gần đây Deep Learning mang lại nhiều thành tựu kĩ thuật nhờ vào sự phát triển của Big Data (Dữ liệu lớn), với tập dữ liệu càng lớn giúp phát triển các mạng Neural Network sâu hơn với khả năng dự đoán chính xác cao hơn (Hình 5.1.2.1). Bắt nguồn từ các loại hình kinh doanh internet truyền thông như tìm kiếm web và quảng cáo. Deep Learning ngày nay đã tạo ra nhiều lợi ích hơn trong cuộc sống hằng ngày. Ví dụ như trong lĩnh vực y tế, việc ứng dụng Deep Learning vào phân tích các hình ảnh X-quang để dự đoán khả năng ung thư của người bệnh, nhận biết các đặng trưng trên cơ thể của một người hoặc các tế bào. Bên cạnh đó Deep Learning còn được ứng dụng khá nhiều trong xe tự hành.



Hình 4.1.1 Quan hệ giữa dữ liệu và khả năng dự đoán của thuật toán.
(Nguồn: Coursera)

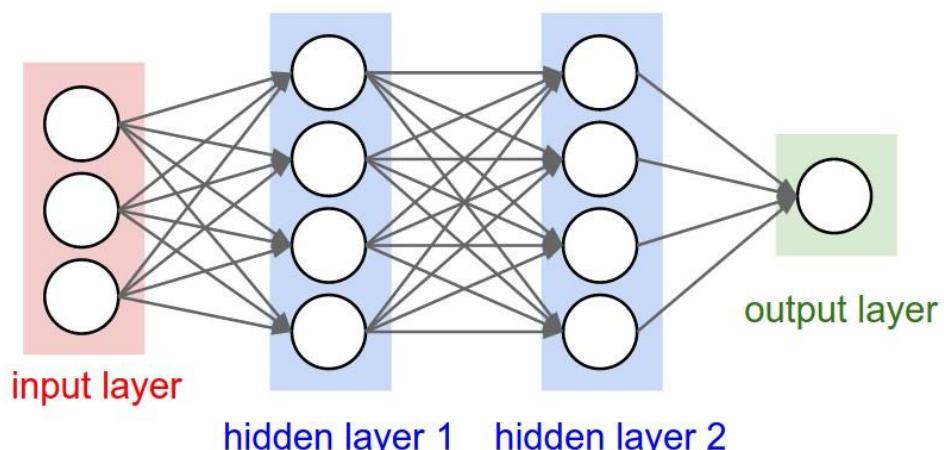
Các ứng dụng khác nhau chúng ta có thể áp dụng các Neuron Network khác nhau:

BẢNG 4.1.1 CÁC ỨNG DỤNG KHÁC NHAU CHÚNG TA CÓ THỂ ÁP DỤNG CÁC NEURON NETWORK KHÁC NHAU

Ứng dụng	Neural Network
Xe tự hành	Custom, Hybrid Neural Network
Xử lý ngôn ngữ tự nhiên, văn bản	Current Neural Network
Nhận diện và phát hiện đối tượng	Convolutional Neural Network
Dự đoán nhà đất. Đưa ra đề suất quản cáo.	Standard Neural Network

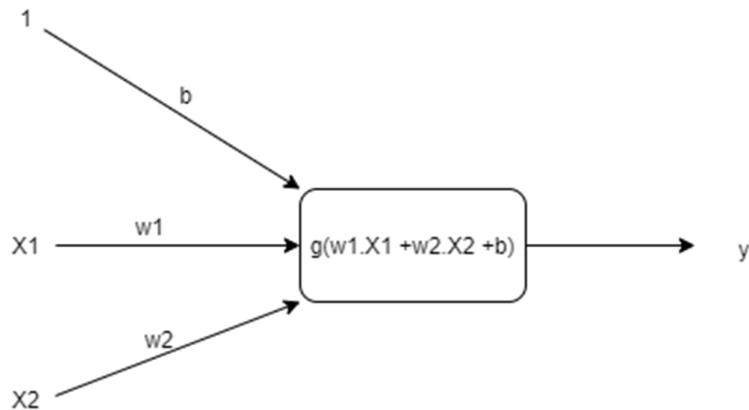
b) Neural Network

- Neural Network là một mạng lưới gồm nhiều lớp được lấy cảm hứng từ neuron người. Ở đó, lớp đầu tiên để đưa các đặc tính của vật cần dự đoán vào được gọi là Input Layer. Và layer cuối cùng mang kết quả dự đoán gọi Output Layer. Một mạng neuron có thể có hoặc không có các lớp ở giữa Input Layer và Output Layer gọi là Hidden Layer, các Hidden Layer này giúp cho tỉ lệ dự đoán chính xác cao hơn tuy nhiên việc huấn luyện cũng tốn nhiều thời gian và dung lượng hơn. Mỗi Layer là tập hợp nhiều node, các node của lớp sau kết nối với toàn bộ các node của lớp trước.



Hình 4.1.2 Kiến trúc mạng Neural Network.

- Mỗi node trong hidden layer và output layer thực hiện các công việc sau: Liên kết với tất cả các node ở layer trước đó với các hệ số w riêng. Mỗi node có 1 hệ số bias b riêng. Từ đó w,b biểu thị mối quan hệ giữa node trước và node sau.



Hình 4.1.3 Mối quan hệ giữa ngõ vào và ngõ ra của một node.

- Node phía trên có đầu vào là X_1, X_2 , trọng số w_1, w_2 . Ngõ ra y là kết quả của một hàm phi tuyến và một hàm tuyến tính. Phương trình của một node mạng có dạng :

$$y = g(W_1X_1 + W_2X_2 + b)$$

- Hàm phi tuyến g còn được gọi là hàm phi kích hoạt (activation) giúp tăng khả năng học của mạng neural.
- Các hàm phi kích hoạt thường được sử dụng là:
 - Sigmoid:** Giá trị ngõ ra được chuyển về trong khoảng $[0,1]$ bằng công thức:

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

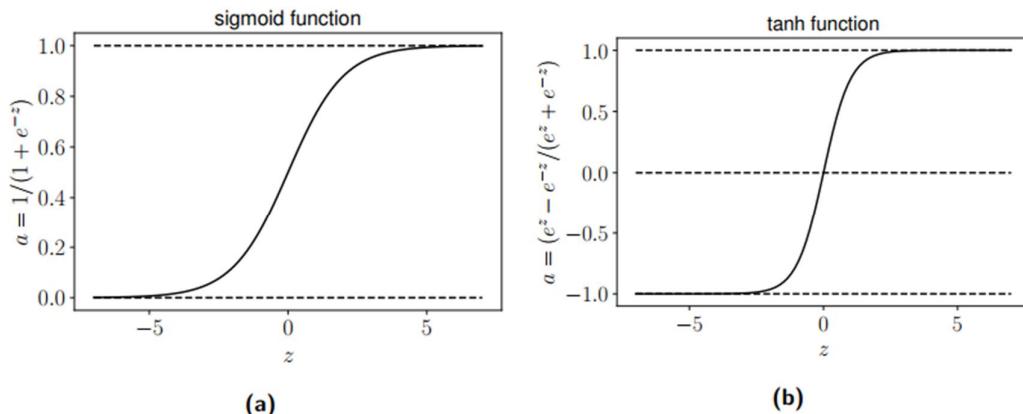
Hàm sigmoid cho ngõ ra có giá trị bằng 1 khi ngõ vào lớn và ngõ ra bằng 0 khi ngõ vào bằng 1. Ở thời gian đầu hàm sigmoid rất được ưa chuộng sử dụng, tuy nhiên sau này vì hàm luôn dương nên khi lấy gradient, hàm sẽ khó hội tụ. Mặt khác hàm sigmoid dễ bị bão hòa và không có gradient khi đầu vào rất nhỏ hoặc rất. Vì vậy, sau này hàm sigmoid chỉ được sử dụng ở những lớp tích chập cuối khi cần tính binary cross-entropy loss.

- Tanh:** Giá trị ngõ ra được chuyển về trong khoảng $[-1,1]$ khiến nó có tính chất tâm không (zero-centered), theo công thức:

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

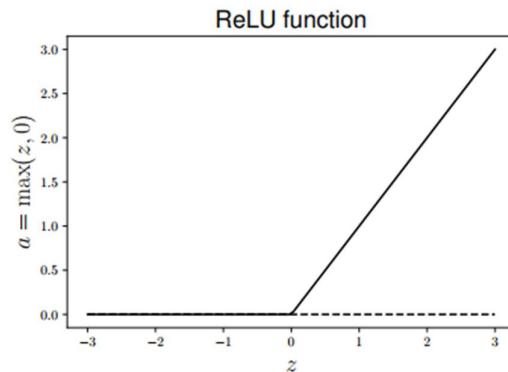
Khi đó, ngõ ra bằng 1 khi ngõ vào lớn và ngõ ra bằng -1 khi ngõ vào nhỏ. Hàm

Tanh với tính chất tâm không giúp các dữ liệu được phân bố quanh điểm 0, tanh khi lấy đạo hàm có cả phần dương và phần âm giúp việc hội tụ trở nên tốt hơn. Tuy nhiên hàm tanh không giải quyết được vấn đề bão hòa khi giá trị ngõ vào quá lớn hoặc quá nhỏ của sigmoid.



Hình 4.1.4 Đồ thị hàm sigmoid (a) và hàm tanh (b)

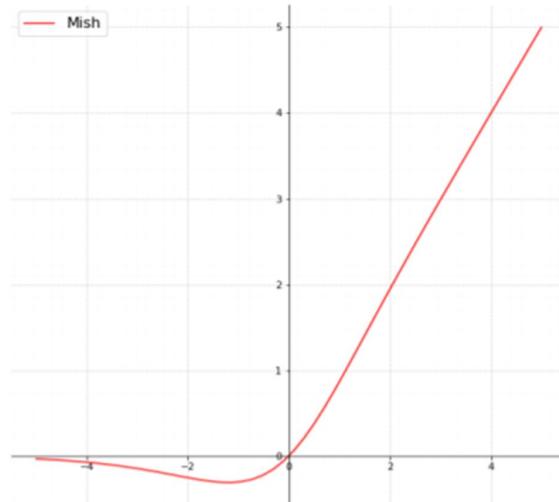
- ❖ **ReLU và leaky ReLU:** Lấy ngưỡng giá trị ở 0 (Thay thế các giá trị âm bằng 0): $g(x) = \max(0, x)$. Hàm ReLU loại bỏ các giá trị âm, việc hội tụ cũng diễn ra nhanh hơn khi không còn bị bão hòa ở hai đầu như hàm Sigmoid và hàm Tanh. Tuy nhiên, với các giá trị âm nhỏ gần 0, việc giữ lại đạo hàm vẫn có giá trị khi lấy Gradient, nhưng lại bị triệt tiêu gây ra hiện tượng “Dying ReLU”. Để khắc phục việc đó, hàm Leaky ReLU ra đời và là biến thể của ReLU: $g(x) = \begin{cases} 0.01x, & x < 0 \\ x, & x \geq 0 \end{cases}$, các giá trị âm lớn gần như xấp xỉ bằng 0 trong khi các giá trị âm nhỏ vẫn mang giá trị khi đạo hàm. Nhưng cả hai hàm ReLU và Leaky ReLU lại không có đạo hàm tại 0.



Hình 4.1.5: Đồ thị hàm ReLU

- ❖ **Mish:** Mish là hàm kích hoạt được sử dụng trong YOLOv4. Mish được tác giả cho rằng tốt hơn các hàm kích hoạt khác nhờ việc giữ được tính chất của hàm ReLU giúp hội tụ nhanh chóng hơn, giữ được một phần gradient âm cho phép model học tốt hơn như hàm Leaky ReLU, mà khắc phục được việc không có đạo hàm tại 0 của hai hàm trên, Mish có đạo hàm tại mọi giá trị. Hàm Mish được biểu diễn bởi công thức:

$$g(x) = x \cdot \tanh(\ln(1+e^x))$$



Hình 4.1.6 Đồ thị hàm Mish

4.2) Convolutional Neural Network

4.2.1) Tổng quát

- Convolutional Neural Network (CNN) hay còn được gọi là mạng Nơ-ron tích chập là một trong những mô hình của Deep Learning. Tác dụng của thuật toán này chính là tạo ra những hệ thống thông minh, có sự phản ứng với độ chính xác cao. Ví dụ như

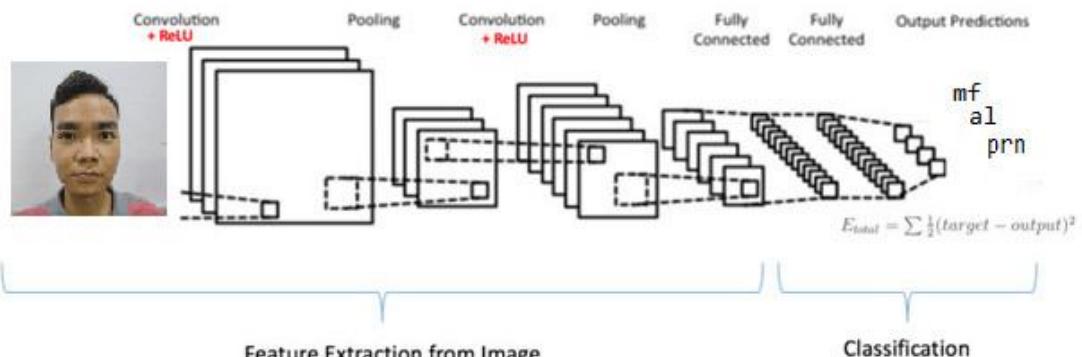
Facebook, Google,... đã đưa vào sản phẩm của mình chức năng nhận diện khuôn mặt,... Ứng dụng cơ bản nhất của thuật toán này là phân lớp, tức là phân biệt hoặc là cái này hoặc là cái kia, tức là khi đưa hình ảnh vào máy tính, nó sẽ là các điểm ảnh hai chiều và điều CNN thực hiện đó là khi các điểm ảnh thay đổi thì máy tính vẫn biết được đó là hình ảnh gì.

- Đầu vào của CNN được chia thành 3 chiều : rộng, cao, sâu. Các Nơ-ron trong mạng không liên kết hoàn toàn với toàn bộ Nơ-ron kế đến mà chỉ liên kết tới một vùng nhỏ. Cuối cùng, một tầng đầu ra được tối giản thành vec-tor của giá trị xác suất.

- CNN gồm 2 thành phần:

- Phần tầng ẩn hay phần rút trích đặc trưng: Trong phần này, mạng sẽ tiến hành tính toán hàng loạt phép tích chập (Convolutional layer) và phép hợp nhất (pooling) để phát hiện các đặc trưng.

- Phần phân lớp: Tại phần này, một số lớp các liên kết đầy đủ (Fully Connected) sẽ đóng vai trò như một bộ phân lớp các đặc trưng đã rút trích trước đó. Tầng này sẽ đưa ra xác suất của một đối tượng trong hình.



Hình 4.2.1 Cấu trúc mạng CNN

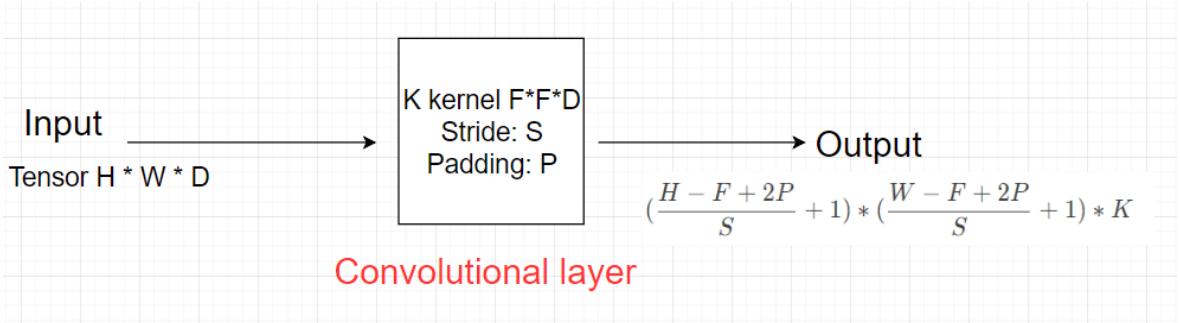
4.2.2) Lớp tích chập (Convolutional Layer).

- Lớp tích chập là một mô hình nhận dạng và phân loại hình ảnh trong CNN. Trong đó máy tính xem ảnh đầu vào là một mảng pixel bao gồm các thông số HxWxD (H: Chiều cao, W: Chiều rộng, D: Độ dày). Với mỗi ảnh đầu vào máy sẽ thực hiện tích chập bằng cách trượt các mặt nạ (bộ lọc) Kernels hoặc Gausse tuần tự lên trên từng vùng ảnh. Tại mỗi vị trí, ta tiến hành phép nhân ma trận và tính tổng để đưa vào feature map (bản đồ đặc tính). Feature map này chính là một ma trận 3 chiều, trong đó bao gồm những con số (parameter).

- Ở lớp đầu tiên, các feature map này dùng để phát hiện các cạnh (thẳng đứng, nằm ngang hoặc cạnh nghiêng) trên những vùng ảnh mà nó thực hiện tích chập. Ở

các lớp tiếp theo, nó có thể phát hiện được các đường cong, gấp khúc trên ảnh. Càng sử dụng nhiều lớp convolution thì càng phát hiện được nhiều các đặc trưng của ảnh. Ví dụ như các đường nét trên khuôn mặt hoặc của một vật thể nào đó. Dưới đây là một số ví dụ về các lớp layer.

- Với mỗi kernel khác nhau ta sẽ học được những đặc trưng khác nhau của ảnh, nên trong mỗi convolutional layer ta sẽ dùng nhiều kernel để học được nhiều thuộc tính của ảnh. Vì mỗi kernel cho ra output là 1 matrix nên 1 kernel sẽ cho ra 1 output matrix. Ta kết hợp k output matrix này lại thành 1 tensor 3 chiều có chiều sâu k.



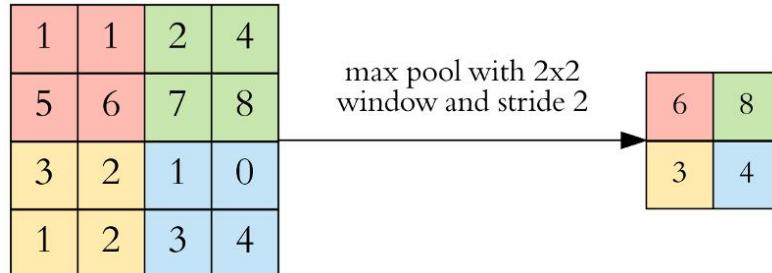
Hình 4.2.2. Các thông số của conv layers

- Tại khối tích chập, các thông số được sử dụng bao gồm:
- Stride (S): Số pixel bạn dịch chuyển ở mỗi lần trượt.
- Padding (P): Tăng thêm một số hàng và cột (thường có giá trị bằng 0) để tránh việc giảm kích thước ở output cũng như để lấy thông tin ở các pixel ngoài cùng. Có 2 loại padding là “valid” tức là padding =0 và “same” tức là padding để kích thước ngõ ra bằng kích thước ngõ vào.
 - F : kích thước của mỗi filter là F^*F .
 - D : là số chiều của ảnh.
 - K là số filter sử dụng trong một lớp.

4.2.3) Lớp tổng hợp (Pooling layer).

- Pooling layer thường được dùng giữa các convolutional layer, để giảm kích thước dữ liệu quan trọng nhưng vẫn giữ được các đặc trưng quan trọng của ảnh. Qua đó giảm khối lượng tính toán trong model.
- Có 2 loại thông dụng:
- Max pooling: Trích suất giá trị lớn nhất của một pooling window.
- Average pooling: Lấy giá trị trung bình của một pooling window.

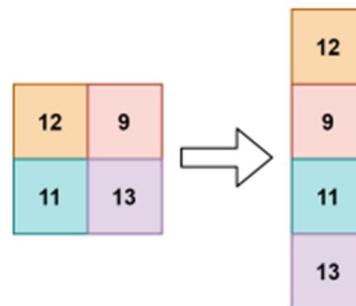
- Trong YOLO người ta thường sử dụng max-pooling vì độ hiệu quả mà nó mang lại. Max-pooling hoạt động gần giống với lớp tích chập (Convolution). Mỗi khi cửa sổ max-pooling trượt qua nó sẽ lấy giá trị lớn nhất của feature map trong cửa sổ trượt.



Hình 4.2.3 Thuật toán max-pooling

4.2.4) Lớp kết nối đầy đủ (*Full connect layer*).

- Sau khi ảnh đầu vào được đưa qua nhiều lớp tích chập và lớp tổng hợp thì model đã học được tương đối nhiều các đặc điểm của ảnh (các đường cong, nghiêng đặc trưng của đối tượng nhận dạng: các cạnh, đường cong trên khuôn mặt ...). Khi ấy tensor của output cuối cùng, kích thước $H * W * D$ sẽ được chuyển về 1 vector kích thước là $(H * W * D)$.



Hình 4.2.4 Chuyển lớp tổng hợp về một vector.

- Sau khi chuyển lớp tổng hợp về một vector *với* chiều dài $(H * W * D)$, mỗi điểm trên vector sẽ được liên kết với toàn bộ output của model giống như một lớp mạng Neural Network bình thường. Lớp cuối cùng này có nhiệm vụ đưa ra các dự đoán theo nhu cầu của người sử dụng.

4.2.5) Batch Normalization

- Trong quá trình training, có thể xảy ra hiện tượng gradients không ổn định do gradients trước và sau khi di chuyển qua 1 layers không giống nhau. Đây là vấn đề liên quan đến phân phối của inputs của các layers.
- Batch Normalization học những giá trị trung bình và phương sai của đầu vào sau đó chuẩn hóa đầu vào để có giá trị trung bình là 0 và phương sai là 1. Việc chuẩn hóa

dữ liệu đầu vào giúp cho quá trình huấn luyện nhanh hơn (hàm chi phí hội tụ nhanh hơn) so với việc không chuẩn hóa.

- Một lợi ích được nhận định của batch normalization là huấn luyện nhanh hơn.
- Thuật toán chuẩn hóa cụ thể là:

Input: Giá trị của x trên mini-batch: $B = \{X_1, X_2, \dots\}$;

Tham số được học là: γ, β

Output: $\{y_i = BN_{\gamma, \beta}(x_i)\}$

Mini -batch mean

$$\mu_B \leftarrow \frac{1}{m} \sum_{i=1}^m x_i$$

//Mini-batch giá trị

$$\sigma_B^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$$

// Normalize

$$\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

//Scale and shift

$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma, \beta}(x_i)$$

(Phép biến đổi Batch normalization)

- Các kí hiệu được đề cập trong bảng tóm tắt kí hiệu

4.2.6) Thuật toán YOLO (You Only Look Once)

• Ở phần trước, việc sử dụng model CNNs chỉ có thể để nhận dạng và phân loại các vật thể khác nhau trên từng bức ảnh. Trên thực tế, nhu cầu về việc phân loại nhiều vật thể trên một khung hình lại phổ biến hơn. Để thực hiện mục đích đó, nhiều thuật toán được đề suất như R-CNN series (R-CNN, Fast R-CNN, Faster R-CNN), YOLO series (YOLOv1, YOLOv2, YOLOv3, mini-YOLOv3 ...). Dựa trên mục đích thực hiện đề tài, nhóm đã lựa chọn YOLOv4 để triển khai sau khi qua quá so sánh và đánh giá tính chính xác, tốc độ học của các thuật toán.

• YOLO được viết tắt từ "You Only Look Once", tạm dịch là "Bạn chỉ nhìn một lần". Thuật toán YOLO xem bài toán phát hiện vật thể là một vấn đề hồi quy duy nhất trên toàn bộ bức ảnh, trực tiếp từ các pixel của ảnh thành các ô dự đoán (bounding box) cùng với xác suất phân loại vật thể thay vì phải thực hiện nhiều bài toán hồi quy cho từng vùng vật thể như các thuật toán R-CNN series. Như vậy việc chỉ sử dụng 1

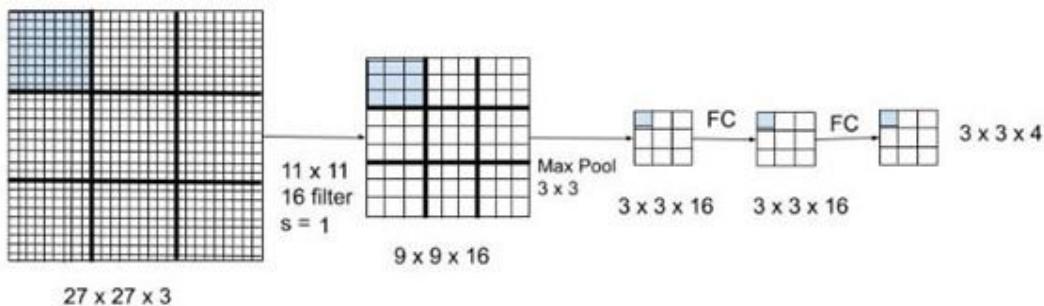
bài toán hồi quy duy nhất cho toàn bộ ảnh, thuật toán YOLO giúp giảm số lượng phép toán, tăng tốc độ xử lý khi đó có thể đáp ứng bài toán thời gian thực tốt hơn so với các thuật toán R-CNN

4.2.7) Hệ thống Grid (Grid system)

- Ý tưởng của việc thực hiện Grid System xuất phát từ việc thay vì thực hiện các lớp Fully Connected ở cuối mô hình thì các lớp này sẽ được chuyển thành các lớp tích chập như các lớp phía trước trong bài toán phân loại vật thể. Khi đó ở lớp cuối cùng thay cho lớp fully connected có kích thước (number of classes , 1) thành một lớp tích chập có kích thước (1, 1, number of classes)

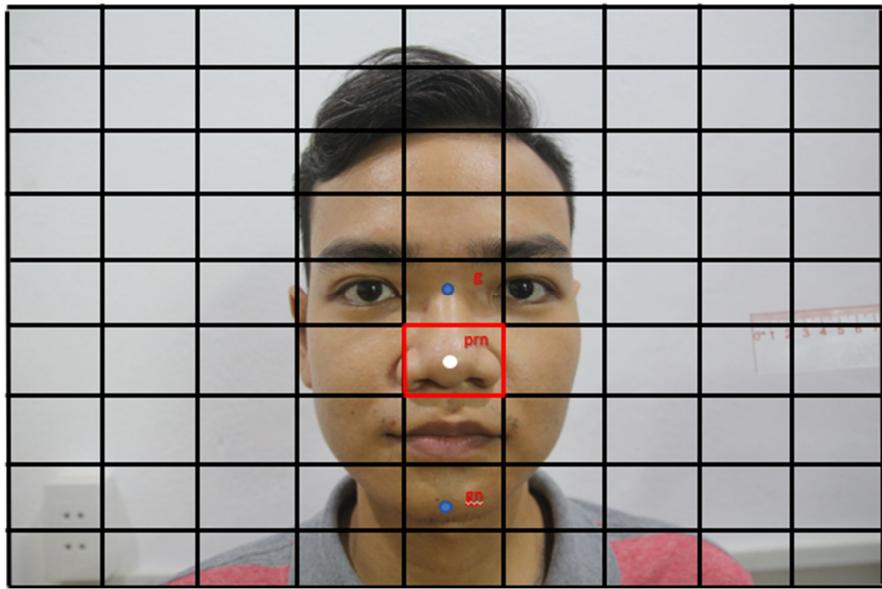
- Việc chuyển đổi này không làm ảnh hưởng đến kết quả dự đoán mà còn thể hiện được chúng ta hoàn toàn có thể phân loại vật thể bằng các phép toán tích chập với kết quả dự đoán nằm ở lớp tích chập cuối cùng và còn ưu việt hơn khi vẫn giữ được vị trí của vật thể.

- Cho một nút mạng như hình ở dưới, với 9 grid (các ô có đường kẻ đậm) được chia ra trong ảnh input ($27 \times 27 \times 3$), vậy tương ứng mỗi grid sẽ chứa ($9 \times 9 \times 3$) pixel. Giả sử vật cần phát hiện ở grid màu xanh, khi qua quá trình convolution và full connect layer, vị trí grid chứa vật vẫn không đổi trên bản đồ đặc tính (feature map).



Hình 4.2.7a Tính bảo toàn vị trí và đặc tính của vật khi sử dụng hệ thống Grid

- Qua đó ta có thể thấy khi sử dụng lớp tích chập ở lớp cuối cùng, ta vừa có thể giữ được các đặc tính của vật, vừa có thể xác định được vị trí của vật thể. Tại mỗi grid sẽ mang 3 thông tin chính bao gồm: Grid có chứa đối tượng cần nhận dạng nào không, tọa độ của các grid bounding box (tọa độ (x,y) tâm của vật, chiều dài, chiều rộng của bounding box), xác suất vật được nhận dạng thuộc class nào.



Hình 4.2.7b : Mô tả ma trận grid

- Ta xem hình 4.2.7b ở trên, giả sử ta cần nhận dạng đối tượng là điểm prn, ta có thể thấy đối tượng người nằm trong một grid (màu đỏ) máy sẽ xác định grid chứa tâm của vật thể. Tại grid đó sẽ chứa các thông tin như hình dưới.

$$y = [p_c, b_x, b_y, b_h, b_w, c_1, c_2, c_3, \dots]$$

- Với p_c là xác suất của vật thể có xuất hiện trong grid hay không.
- 4 giá trị tiếp theo theo lần lượt là tọa độ tâm (x,y) của vật thể, chiều rộng và chiều dài của vật thể c_1, c_2, c_3 là class của vật nhận dạng được.
- Lấy ví dụ ở hình trên, xét vị trí grid màu đỏ chứa tâm đối tượng cần nhận dạng và có tất cả 3 class ta cần nhận dạng theo thứ tự là prn, n, gn. Ta có:

$$y = [1, 120, 240, 80, 150, 1, 0, 0]$$

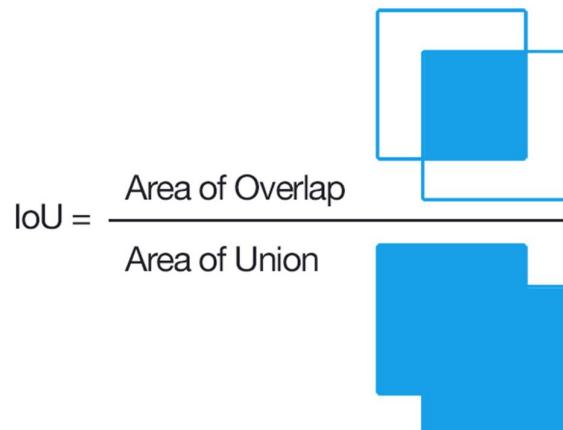
- Ở $y[0] = 1$ có ý nghĩa là ô grid có chứa vật, $y[2:4]$ lần lượt là tọa độ tâm (x,y), chiều rộng, chiều dài của bounding box, $y[5:7]$ là vị trí của các class ta đang xét trong trường hợp này $y[5] = 1$ là hệ thống đang nhận dạng grid này thuộc class prn, $y[6:7] = [0,0]$, miêu tả grid này không có đối tượng là điểm n và gn.
- Nếu xét grid không chứa vật thể y sẽ được gán $y = [0, x, x, x, x, x, x, x]$ tức là không có vật thể ở grid đó và các giá trị còn lại không cần quan tâm.

❖ Khái niệm về IoU (Intersection over Union) và Non – maximum suppression.

- Chỉ số IoU (Intersection over Union) cho ta biết tỉ lệ trùng nhau của một bounding box (box máy dự đoán) và ground – truth bounding box (box mà ta dán nhãn bằng tay)

$$IoU = \frac{box A \cap box B}{box A \cup box B}$$

- Trong đó $box A \cap box B$ là phần giao nhau (Intersection) của hai box, tức là lấy phần chung của A và B. Xem hình 5.2.7.3 .
- $Box A \cup box B$ là phần hợp (Union) của hai box, tức là tổng diện tích của hai box A và B. Xem hình 5.2.7.3



Hình 4.2.7c : Tính toán tỉ lệ IoU

- Trong quá trình traning, hệ thống sẽ đưa ra nhiều bounding box khác nhau để dự đoán (mỗi bounding box có xác suất khác nhau). Để tìm ra IoU phù hợp nhất ta sử dụng thuật toán Non – maximum suppression. Thuật toán sẽ loại các bounding box có tỉ lệ dự đoán thấp và giữ lại 1 bounding box có tỉ lệ IoU cao nhất. Quy trình của nó diễn ra như sau:

Bước 1: Loại bỏ tất cả các bounding box có tỉ lệ p_c thấp hơn ngưỡng overlap threshold, việc này giúp cho các grid không chứa vật thể có tỉ lệ p_c thấp sẽ được loại bỏ .

Bước 2: Chọn các bounding box có xác suất xuất hiện vật thể cao nhất.

Bước 3: Nếu có nhiều hơn một bounding box trong một grid thì khi đó ta sẽ so sánh tỉ lệ IoU giữa các bounding box với nhau và chọn ra một bounding box có tỉ lệ lớn nhất.

4.2.8) Gán nhãn dữ liệu ảnh.

- Trong các bài toán deep learning để thực hiện gồm có hai quá trình cho “máy học” và quá trình “kiểm tra”. Để máy có thể “tự học” thì chúng ta cần hướng dẫn cho nó, việc hướng dẫn này chúng ta gọi là quá trình dán nhãn vật thể ta muốn máy nhận dạng. YOLO cũng là một thuật toán trong deep learning, vì vậy cũng sẽ có quá trình dán nhãn (label) để dạy cho máy học, khi dán nhãn, chúng ta sẽ xác định vị trí tâm,

chiều dài và chiều rộng của label, và cuối cùng là các lớp (class) chúng ta muốn máy nhận dạng. Khi đó y sẽ có dạng:

$$y = [p_c, b_x, b_y, b_h, b_w, c_1, c_2, c_3, \dots]$$

- Tuy nhiên, YOLO không thể nhận diện hai vật thể trong cùng một grid. Để giải quyết vấn đề trên ta có hai cách:

• Cách 1: Tăng số lượng grid trên hình đầu vào lên cho tới khi tâm của hai vật nằm ở hai grid khác nhau. Cách này chỉ hiệu quả với các vật nhỏ, hai tâm không sát nhau. Lý do khi chia các grid ra quá nhỏ sẽ làm tăng độ khó của lớp mạng (các đặng trưng của ảnh trong một grid bị ít đi), làm tăng thời gian nhận dạng và tính chính xác bị giảm đáng kể.

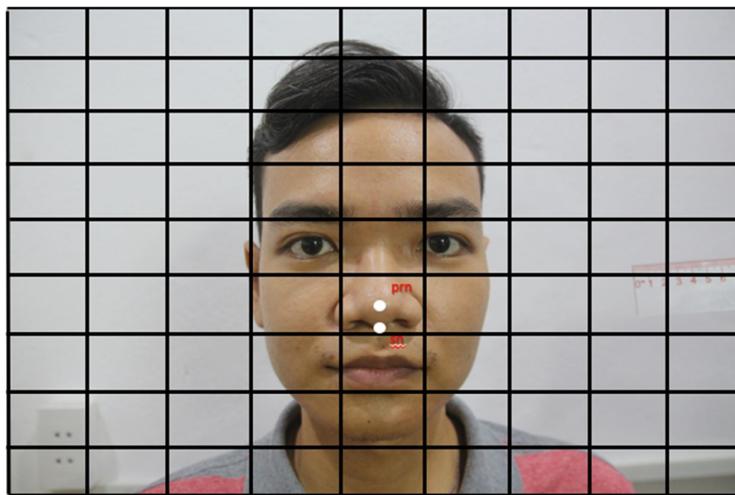
• Cách 2: Ta nhận thấy rằng phương trình y ở trên chỉ có thể nhận dạng cho một vật thể duy nhất, ý tưởng ở đây là bỏ các tham số của vật cần nhận diện vào chung một phương trình y. Khi đó ta có:

$$y = [p_{c1}, b_{x1}, b_{y1}, b_{h1}, b_{w1}, c_1, c_2, c_3, \dots, p_{c2}, b_{x2}, b_{y2}, b_{h2}, b_{w2}, c_1, c_2, c_3, \dots]$$

- Xét ví dụ với hình 5.2.8 bên dưới, điểm prn và điểm sn đều nằm cùng chung 1 grid. Khi đó, y sẽ được gán như sau:

$$y = [1, 120, 20, 20, 120, 1, 0, 1, 90, 50, 90, 20, 0, 1]$$

• Tức là ở grid này có xuất hiện điểm sn ($y[0] = 1$), (120,20,20,120) là 4 thông số anchor box của điểm sn, ($y[5]=1, y[6]=0$) để phân loại cho anchor box này là điểm sn chứ không phải điểm prn, tương tự $y[7]=1$ có nghĩa grid này cũng có điểm prn, 4 thông số tiếp theo để xác định tọa độ anchor box cho điểm prn, ($y[12]=0, y[13]=1$) để phân loại anchor box này là điểm prn chứ không phải điểm sn. Như vậy, với cách 2 ta sẽ ghép y của 2 hay nhiều vật thể nằm trong 1 grid lại thành một, $y[0:6]$ để xác định cho việc phát hiện điểm sn, và $y[7:13]$ để xác định cho điểm prn. Tuy nhiên với cách thứ 2, nếu ta ghép càng nhiều tốc độ xử lý càng lâu do càng có nhiều phép toán phép thực hiện, vì vậy không nên quá lạm dụng cách này mà nên phối hợp hài hòa với cách thứ 1, tăng số grid phải chia lên.



Hình 4.2.8: Tâm của 2 vật thể cùng nằm trong 1 grid.

4.2.9) Hàm mất mát

- Sau khi gán nhãn cho toàn bộ tập dữ liệu, các mẫu dữ liệu sẽ được đưa qua mạng CNN (tất cả ảnh sẽ đưa vào mạng CNN) để thực hiện việc học các tham số. Trong quá trình này, các hàm mất mát sẽ được tính toán. Đối với bài toán nhận diện vật thể sẽ cần tính toán 3 hàm mất mát.

- Hàm mất mát phân loại (Classification Loss Function):

$$L_{classification} = \sum_{i=0}^{S^2} \prod_i^{obj} \sum_{c \in class} (p_i(c) - \hat{p}_i(c))^2$$

- Hàm mất mát vị trí (Localization Loss Function): Được sử dụng để tính toán độ sai số giữa các bounding box dự đoán với các anchor box, cải thiện Localization Loss sẽ giúp việc phát hiện vật thể trở nên chính xác hơn:

$$L_{localization} = \sum_{i=0}^{S^2} \sum_{j=0}^B \prod_{ij}^{obj} [(offsetx_i - \hat{offsetx}_i)^2 + (offsety_i - \hat{offsety}_i)^2 + (width_i - \hat{width}_i)^2 + (height_i - \hat{height}_i)^2]$$

- Hàm mất mát dự đoán (Confidence Loss Function): thể hiện sai số giữa dự đoán của bounding box với nhãn thực tế:

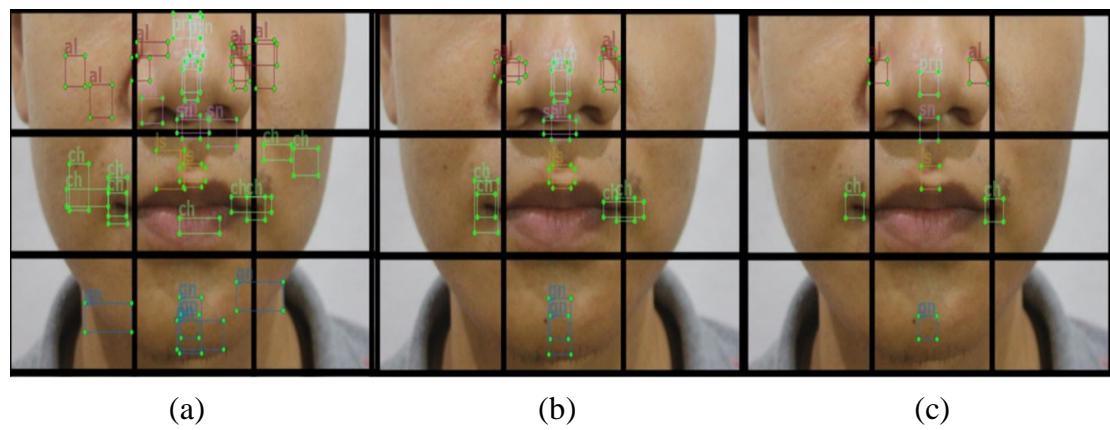
$$L_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^B \prod_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobject} \sum_{i=0}^{S^2} \sum_{j=0}^B \prod_{ij}^{noobj} (C_i - \hat{C}_i)^2$$

- Hàm mất mát tổng quát sẽ bằng tổng 3 hàm mất mát phía trên:

$$L_{total} = L_{classification} + L_{localization} + L_{confidence}$$

Ngõ ra

• Ở ngõ ra, mỗi grid sẽ thực hiện dự đoán 2 bounding box có p_c cao nhất. Loại bỏ tất cả bounding box có p_c thấp trên toàn bộ bức ảnh. Và cuối cùng thuật toán Non-max suppression sẽ được thực hiện để giữ lại bounding box chính xác nhất cho từng vật thể.



Hình 5.2.9: Thuật toán được áp dụng ở ngõ ra

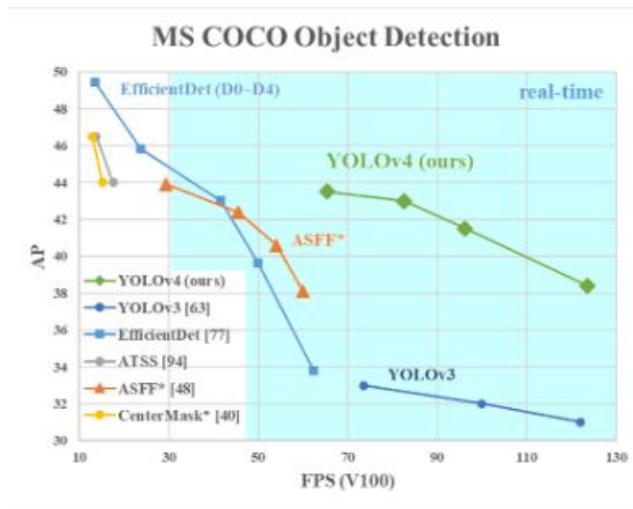
(a): Với mỗi grid, lấy 2 bounding box có p_c cao nhất.

(b): Loại bỏ tất cả các bounding box có p_c thấp trên toàn bộ bức ảnh, bước này giúp loại bỏ các grid không chứa vật thể.

(c): Áp dụng non-max suppression để chỉ chọn ra bounding-box cuối cùng có xác suất dự đoán cao nhất.

4.3 Thuật toán YOLOv4

- Thuật toán YOLOv4 về cơ bản cũng thừa kế các phương pháp cơ bản của các YOLO, tuy nhiên YOLOv4 áp dụng một số thuật toán phát hiện vật thể nhanh, tối ưu hóa các phép toán thực hiện song song giúp tăng tốc độ nhận diện và tăng độ chính xác.



Hình 4.3 So sánh tốc độ xử lý và độ chính xác của YOLOv4 với các thuật toán khác trong cùng tập dữ liệu MS COCO.

(Có thể thấy so với YOLOv3 với cùng một FPS (Frame per Second) YOLOv4 cho độ chính xác (AP) cao hơn hẳn.)

4.4) Cấu trúc nhận diện vật thể của YOLOv4

Cấu trúc nhận diện vật thể của YOLOv4 thường có 3 phần:

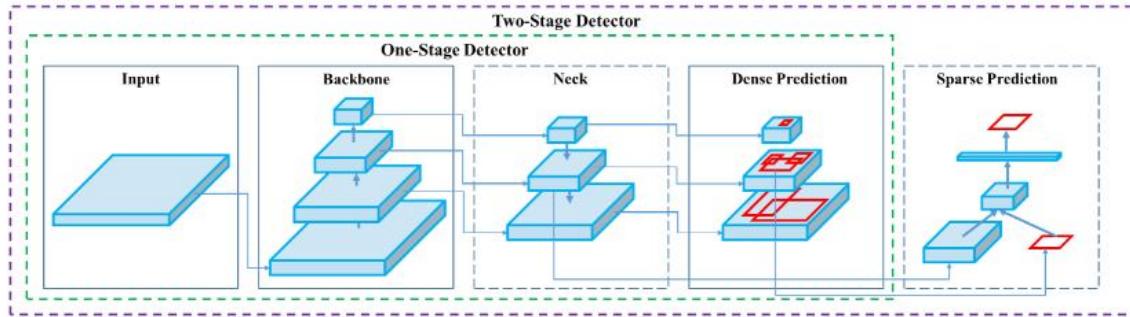
- Backbone: Backbone là 1 mô hình pre-train của 1 mô hình học chuyển (transfer learning) khác để học các đặc trưng và vị trí của vật thể. Các mô hình học chuyển thường là VGG16, ResNet-50, ... Mô hình học chuyển được áp dụng trong YOLOv4 là CSP Darknet53

- Head: Phần head được sử dụng để tăng khả năng phân biệt đặc trưng để dự đoán class và bounding-box. Ở phần head có thể áp dụng 1 tầng hoặc 2 tầng:

- Tầng 1: Dense Prediction, dự đoán trên toàn bộ hình với các mô hình RPN, YOLO, SSD,...

- Tầng 2: Sparse Prediction dự đoán với từng mảng được dự đoán có vật thể với các mô hình R-CNN series, ...

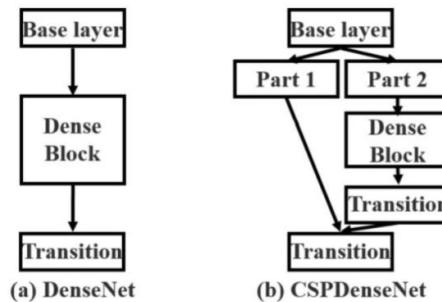
- Neck: Ở phần giữa Backbone và Head, thường có thêm một phần Neck. Neck thường được dùng để làm giàu thông tin bằng cách kết hợp thông tin giữa quá trình bottom-up và quá trình top-down (do có một số thông tin quá nhỏ khi đi qua quá trình bottom-up bị mất mát nên quá trình top-down không tái tạo lại được). Các mô hình được dùng trong quá trình Neck của YOLOv4 là SPP, PAN



Hình 4.4.1: Cấu trúc nhận diện vật thể của YOLOv4

- Backbone:

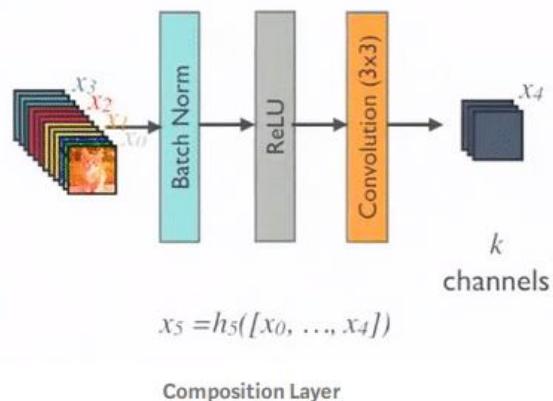
- Với backbone, YOLOv4 áp dụng mô hình CSPDarkNet53 vì theo tác giả mô hình có độ chính xác cao hơn so với các mô hình ResNet thông thường, và tăng khả năng phân loại nhờ vào hàm Mish. CSP-Net về cơ bản cũng tương tự như mô hình DenseNet nhưng có một số khác biệt. CSP(Cross-stage-partial-connection) chia thông tin thành 2 phần bằng nhau, một phần được đưa thẳng vào lớp chuyển kế tiếp (Transition block), phần còn lại được đưa vào dense block (được nhắc đến ở phần sau) trước khi đưa vào lớp chuyển kế tiếp. Chính nhờ phần đưa vào dense block giúp giữ thông tin của các lớp phía trước, và phần đưa thẳng vào Transition block giúp giảm số tham số cần tính toán.



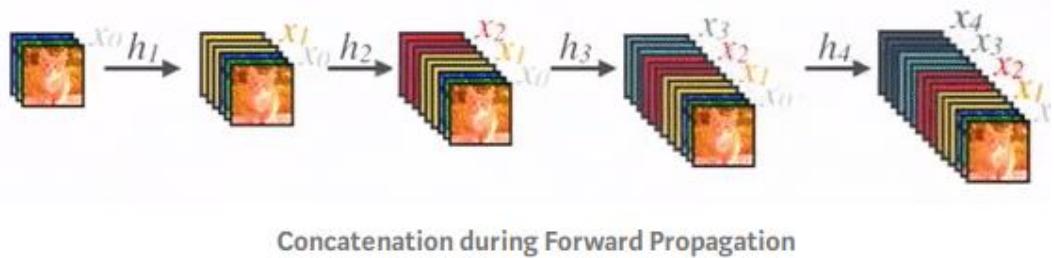
Hình 4.4.2: Sự khác biệt giữa mạng DenseNet và mạng CSPDenseNet.

- DenseNet là một trong những mô hình mạng mới nhất cho kỹ thuật nhận diện vật thể được công bố vào năm 2016. DenseNet gồm 2 khối, trong đó khối chính là Dense Block và 1 khối chuyên. Dense Block là một kĩ thuật mở rộng của skip-connection. Dense Block giúp mô hình trở nên phức tạp hơn làm tăng khả năng học của mô hình, từ đó tăng độ chính xác mà lại giảm được 1 nửa số tham số so với mô hình ResNet trước đó. Tương tự như ResNet, DenseNet cũng giúp giảm overfitting hay vanishing gradient.

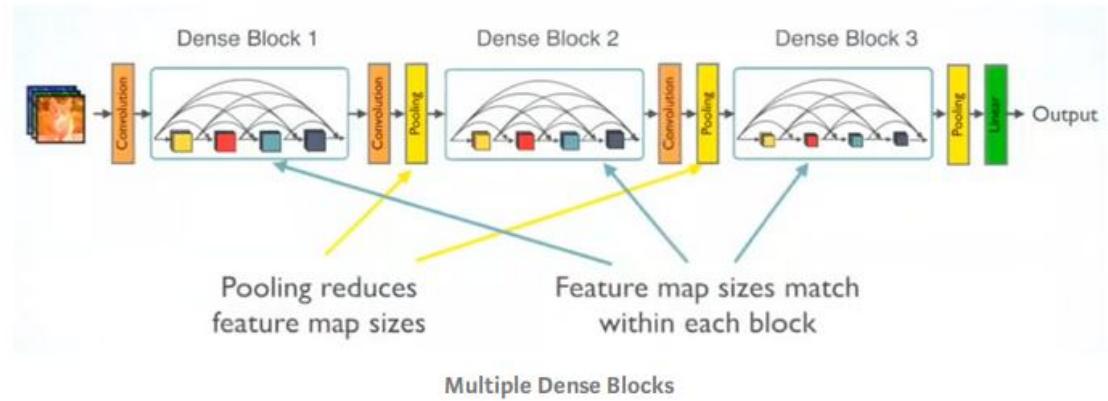
- Lớp tích chập đầu tiên được đưa vào DenseNet sẽ thực hiện Batch-Normalization, ReLU và lấy tích chập với 1 filter có kích thước 3×3 để lấy được lớp tích chập thứ 2 . Lớp tích chập thứ 2 này sẽ tiếp tục thực hiện ghép chồng (concatenate) với lớp tích chập thứ 1, đây cũng là điểm mới của DenseNet so với ResNet khi ResNet sẽ chỉ lấy element-wise, sẽ làm giảm đáng kể số tham số phải học so với ResNet. Quá trình trên tiếp tục lặp lại với lớp tích chập thứ 2, thứ 3,... Kết thúc quá trình, lớp tích chập cuối cùng sẽ được đưa vào lớp chuyển để thực hiện lấy tích chập hoặc max-pooling và lại đưa vào một Dense Block mới.



Hình 4.4.3: Quá trình lấy Batch-Norm, ReLU, tích chập với filter 3×3 của 1 lớp



Hình 4.4.4 Quá trình ghép chồng lớp tích chập phía trước vào lớp tích chập phía sau



Hình 4.4.5 Mô hình DenseNet với 3 khối Dense Block và các lớp chuyển chèn giữa

- Mô hình Darknet-53 gồm 53 tầng tích chập đã được giới thiệu trong YOLOv3

Type	Filters	Size	Output
Convolutional	32	3×3	256×256
Convolutional	64	$3 \times 3 / 2$	128×128
1x	32	1×1	
	64	3×3	
	Residual		128×128
Convolutional	128	$3 \times 3 / 2$	64×64
2x	64	1×1	
	128	3×3	
	Residual		64×64
Convolutional	256	$3 \times 3 / 2$	32×32
8x	128	1×1	
	256	3×3	
	Residual		32×32
Convolutional	512	$3 \times 3 / 2$	16×16
8x	256	1×1	
	512	3×3	
	Residual		16×16
Convolutional	1024	$3 \times 3 / 2$	8×8
4x	512	1×1	
	1024	3×3	
	Residual		8×8
Avgpool		Global	
Connected		1000	
Softmax			

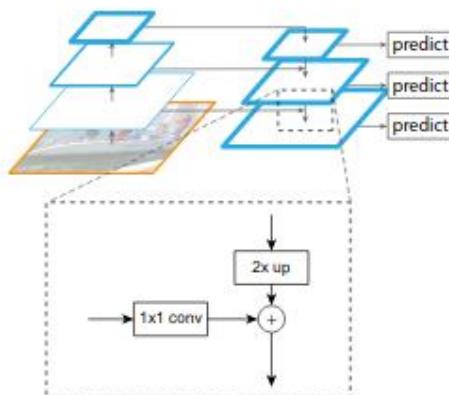
Table 1. Darknet-53.

Hình 4.4.6 Mô hình tích chập Darknet-53 gồm 53 tầng tích chập

- Neck:

- Quá trình backbone, khi đi đáy lên đỉnh (bottom-up stream) giúp việc xác định vị trí của vật thể nhanh và chính xác, tuy nhiên khi đi qua quá trình trên việc các feature map càng ngày càng thu nhỏ, độ phân giải giảm xuống làm cho một số các vật thể

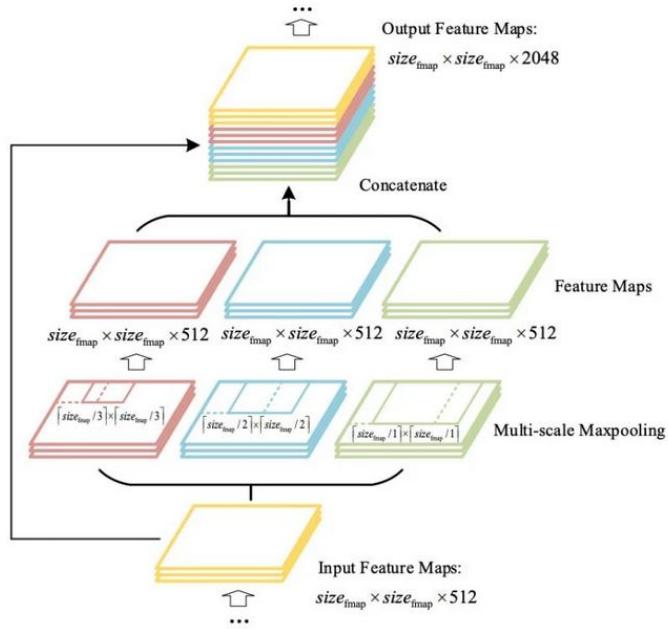
nhỏ qua đó cũng bị biến mất làm mất thông tin. Để khắc phục điều đó trước khi đẩy về head để nhận dạng, Neck thường thực hiện một quá trình ngược lại (top-down stream) tạo ra các reconstruction map để khôi phục lại một số thông tin bị mất trong quá trình bottom-up stream. Nhưng một số thông tin đã bị mất khi khôi phục bằng quá trình top-down cũng không còn hiển thị lại nữa, do đó mạng FPN (Feature Pyramid Network), sẽ tái tạo lại các thông tin bị mất nhờ các kết nối skip-connection giữa các feature map được lấy tích chập 1x1 và reconstruction map tạo ra các feature map mới giàu thông tin giúp việc phát hiện và phân loại vật thể đạt độ chính xác cao hơn.



Hình 4.4.7 : Mô hình FPN.

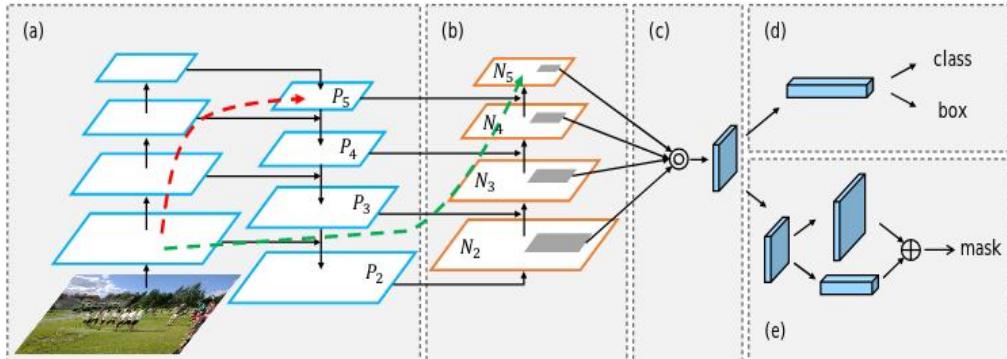
- Trong đó các ô trong khung bottom-up là các feature map trải qua trong quá trình backbone. Các ô trong khung top-down là các reconstruction map. Các ô có viền nét đứt là các feature map sau khi thực hiện skip-connection.

- Trong YOLOv4, tác giả có 2 cải tiến đối với mô hình FPN gồm:
 - Đổi với khối skip-connection giữa feature map và reconstruction map, tác giả sử dụng mô hình YOLO-SPP (Spatial Pyramid Pooling) thay cho các feature map được tích chập 1x1. YOLO-SPP được giới thiệu trong YOLOv3 và tiếp tục được sử dụng trong YOLOv4. YOLO-SPP sử dụng các mạng tích chập max-pooling với các kích thước filter khác nhau, sau khi được lấy tích chập các lớp ngõ ra sẽ được xếp chồng lên nhau. Việc thực hiện max-pooling như vậy giúp giữ được các đặc trưng quan trọng của feature map mà gần như không làm giảm tốc độ xử lý.



Hình 4.4.8: Mô hình YOLO-SPP

- Đối với mô hình đường tổng hợp (Path Aggregation) trước khi đẩy ra Head, tác giả sử dụng mạng PAN (Path Aggregation Network).



Hình 4.4.9: Mô hình mạng PAN

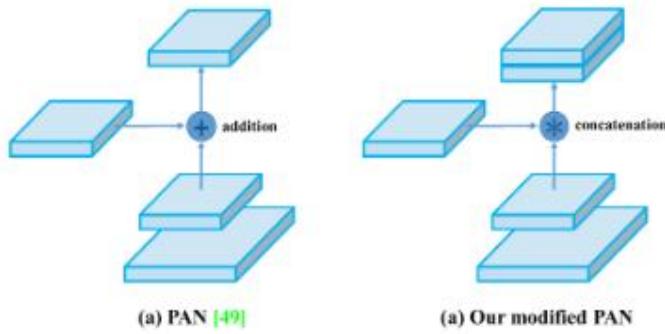
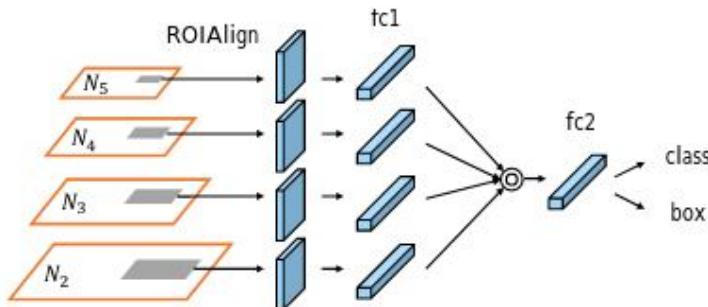


Figure 6: Modified PAN.

Hình 4.4.10: Thay vì thực hiện cộng như mô hình PAN, YOLOv4 thực hiện phép gộp.

- Sau đó, các vật thể sẽ được nhận diện ở từng tầng. Chính vì từng tầng, các vật thể lại được nhận diện lại dẫn đến việc dư thừa dữ liệu. Do đó, với các vùng ROI (Region of Interest - Vùng quan tâm) sẽ được đi qua 1 mạng SPP để lấy tích chập rồi đưa về mạng Fully Connected, rồi thực hiện lấy max đối với các phần tử (element-wise max) để thu được prediction.



Hình 4.4.11 ROI sẽ đi qua mạng SPP để thu được prediction

- Head:

- Ở phần Head để thực hiện dự đoán bounding box và class, YOLOv4 được thực hiện giống như YOLOv3.

- Dự đoán cho bounding box: Sử dụng hồi quy logistic để dự đoán. Kết quả dự đoán bằng 1 nếu bounding box có sự trùng lặp so với anchor box tốt nhất trong các bounding box. Các bounding box khác dù có vượt qua mức ngưỡng nhưng không phải là bounding box tốt nhất cũng sẽ bị gán là 0.

- Phân lớp: Để tăng kết quả dự đoán với bài toán phân loại đa nhãn (multi-label classification), tác giả sử dụng phân lớp logistic độc lập (independent logistic classifier) thay cho hàm softmax. Ví dụ với bài toán phân loại 2 lớp, hàm softmax

cho kết quả predict chỉ là 0,4 và 0,48 trong khi với hàm phân lớp logistic độc lập kết quả này có thể tăng lên 0.85 và 0.8 cho 2 lớp trên.

4.5) Mô hình pre-trained:

- Transfer learning là một phương pháp học máy mà trong đó, một mô hình đã được phát triển cho một task (pretrained network) được tái sử dụng ở một task khác. Phương pháp này mang đến một cách tiếp cận phổ biến và hiệu quả cao trong deep learning khi bạn có một tập dữ liệu vừa và nhỏ. Pretrained network là một mạng đã lưu trước đó, được đào tạo trên một tập dữ liệu lớn (thường là trên một task phân loại ảnh quy mô lớn). Một số mạng pretrained thường được sử dụng như: VGG, ResNet, Inception, Inception-ResNet, Xception....

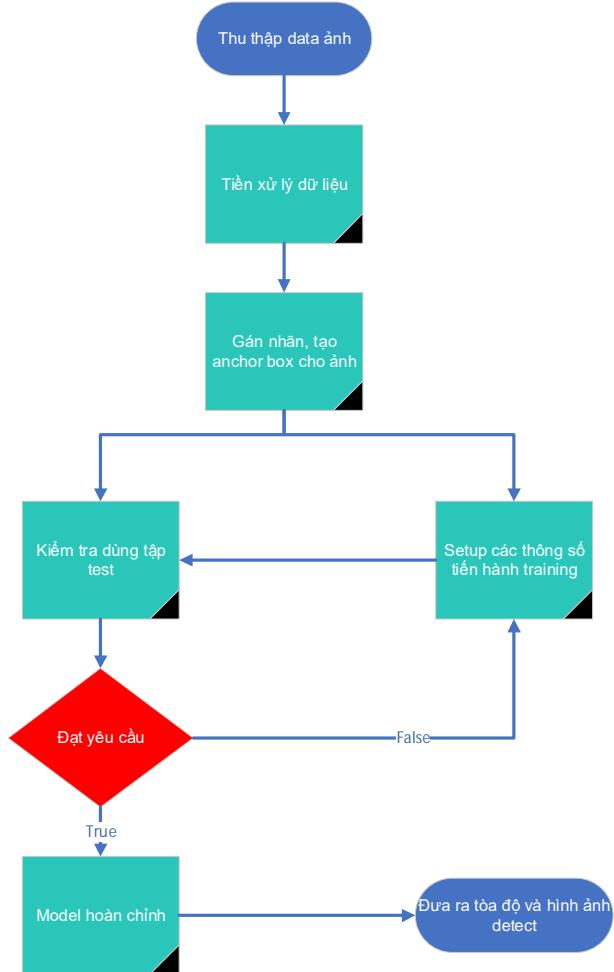
CHƯƠNG 5: NHẬN DẠNG TÍNH TOÁN THÔNG SỐ NHÂN TRẮC

Chương này trình bày quá trình thực hiện, phân loại các vùng nhân trắc sử dụng YOLOv4 (mô hình CNN). Phần 5.2 đưa ra sơ đồ khái của hệ thống nhận dạng. Các phần còn lại thể hiện các bước lấy số liệu, tạo dao diện và cách lưu trữ dữ liệu tính toán.

5.1) Xây dựng mô hình nhận dạng đặc điểm nhân trắc tai, mũi bằng ảnh chụp

• Trong chương này, chúng tôi sử dụng thuật toán YOLOv4 để thực hiện việc phát hiện các đặc điểm nhân trắc của con người. Tập dữ liệu là ảnh được bên thứ 3 cung cấp, ảnh chụp từ máy chụp tự động và xử lý bằng tay được thể hiện ở phần Phản thể hiện rõ quy trình huấn luyện mô hình YOLOv4 trên Google Colab. Phần kiểm tra và đánh giá mô hình đã huấn luyện được.

5.2) Sơ đồ khái tổng quát



- Để hoàn thành việc phát hiện (detect) các điểm nhân trắc sử dụng model YOLO, cần thực hiện các bước sau:

Bước 1: Tìm kiếm và thu thập ảnh của đối tượng cần nhận diện, tiến hành gán nhãn cho từng bức hình

Bước 2: Phân chia dữ liệu thành tập huấn luyện và tập kiểm tra.

Bước 3: Huấn luyện mô hình sử dụng dữ liệu trong tập huấn luyện

Bước 4: Kiểm tra và đánh giá mô hình sử dụng hình ảnh chụp được ở máy chụp tự động.

5.3) Dataset và dán nhãn.

- Để huấn luyện một mô hình deep learning ta cần có tập dữ liệu (dataset), nó là yếu tố tiên quyết để ta huấn luyện mô hình. Dữ liệu ảnh dễ bị ảnh hưởng bởi các tác nhân bên ngoài (ở mỗi thời điểm là khác nhau, vị trí kích thước pixel ảnh, màu sắc ảnh sẽ khác nhau). Nên ta cần có một tập dữ liệu đủ lớn để cho máy có thể nhận biết được các đối tượng ta cần nhận dạng. Làm tăng độ chính xác bên ngoài thực tế

Tập dữ liệu được chia làm 3 phần:

- Phần training: Đây là phần chúng ta để cho máy học các đặc tính của đối tượng cần nhận dạng trên ảnh, vì vậy đây là phần lớn nhất. Để tạo tập training, chúng ta cần dán nhãn các đối tượng, việc dán nhãn giúp ta xác định được tọa độ vị trí đối tượng cần nhận dạng, qua đó tính toán loss function và tỉ lệ IoU, cập nhật các trọng số làm tăng độ chính xác của mô hình.

- Phần validation: Sử dụng để kiểm tra lại tính đúng đắn của mô hình trong quá trình training, thường thì tập validation sẽ lấy khoảng 30% dữ liệu từ tập training để đánh giá độ hiệu quả của mô hình mà không tham gia vào việc cập nhật trọng số. Việc đánh giá độ hiệu quả của phần training sẽ cho ta biết có cần điều chỉnh các thông số parameter lại hay không, lớp mạng có bị high- low bias, high – low varicance hay không.

- Phần test: Đây là tập rác qua trọng dùng để đánh giá mô hình sau khi hoàn tất quá trình training. Nó sẽ thể hiện tính đúng sai dựa trên kết quả thực tế chúng ta mong muốn. Vì lý do đó, tập test không được sử dụng lại dữ liệu của tập training hay validation để tránh hiện tượng overfit hoặc underfit.

Sai số tập training	5%	20%	20%	1.5%
Sai số tập validation	50%	18%	40%	2.25%
Đánh giá mô hình	Mô hình bị high variance (overfitting)	Mô hình bị high bias (underfitting)	Mô hình bị cả high bias và high variance	Mô hình hoạt động ổn định

- Các tập training, validation, test thường được chia với tỉ lệ 80*80/80*20/20 với tập dữ liệu không quá lớn và được chia tỉ lệ 98/1/1 với tập dữ liệu rất lớn (khoảng 1 triệu mẫu).

- Tập dữ liệu gần 2343 tấm ảnh chân dung chụp lần lượt theo 3 hướng nghiêng trái 90 độ, chính diện, nghiêng phải 90 độ. Trong đó có khoảng 1833 tấm chân dung chụp bằng máy chụp tự động và 510 tấm thực hiện phương pháp tăng cường dữ liệu: xoay góc ngẫu nhiên từ 0 độ đến 45 độ.



Hình 5.2a Một số hình ảnh minh họa trong tập dữ liệu train

- Trong đó, tập dữ liệu training có 1500 mẫu ảnh bao gồm (500 hình chụp bên trái, 500 hình chụp bên phải và 500 hình chụp chính diện). Với 29 class (đối tượng cần nhận diện).
- Dưới đây là vị trí các điểm cần nhận dạng nhân trắc trên khuôn mặt của một người.



Hình 5.2b Vị trí các điểm nhân trắc trên khuôn mặt theo hai hướng đối diện và bên phải.

BẢNG 5.2 : TÊN LABEL VÀ VỊ TRÍ CỦA CÁC ĐIỂM NHÂN TRẮC CẦN NHẬN DẠNG.

STT	Tên label	Vị trí label trên khuôn mặt	STT	Tên label	Vị trí label trên khuôn mặt
0	1	Vị trí cao nhất của vầng tai	15	16	Gốc điểm sụn cao nhất trên bờ mặt tai.
1	2	Vị trí thấp nhất của tai	16	17	Điểm đuôi mắt.
2	3	Điểm gần nhất của vầng tai tính từ chân mắt	17	tr	Điểm giữa đường chân tóc.
3	4	Điểm xa nhất của vầng tai tính từ chân mắt	18	g	Điểm gabella (điểm lồi trước nhất của trán).

4	5	Điểm gốc tai trên	19	en	Điểm endocanthion (điểm khói mắt trong).
5	6	Điểm gốc tai dưới	20	n	Điểm nasion (điểm gốc mũi).
6	7	Điểm vân tai cao nhất	21	mf	Điểm maxilofrontale (điểm nền gốc mũi)
7	8	Điểm đầu vành cuồng	22	al	Điểm alare (điểm cánh mũi)
8	9	Điểm hốc tai thấp nhất	23	prn	Điểm pronasale (điểm đỉnh mũi trên da)
9	10	Điểm giữa cuồng vân tai	24	sn	Điểm subnasale (điểm dưới mũi)
10	11	Điểm xun gần nhất của tai (tính từ điểm đuôi mắt)	25	ch	Điểm mép miệng.
11	12	Điểm xun gần xa của tai (tính từ điểm đuôi mắt)	26	ls	Điểm môi trên.
12	13	Điểm xun cao nhất trên bờ mặt tai.	27	gn	Điểm cầm
13	14	Điểm vành tai trên	28	k	Điểm gù xương mũi.
14	15	Điểm vành tai dưới			

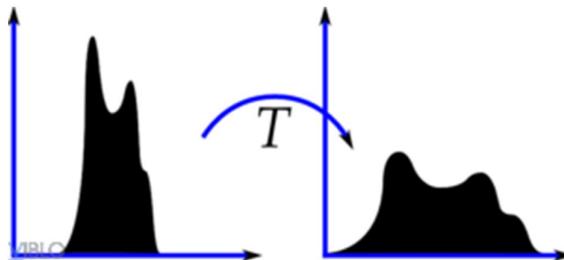
5.4) Tiền xử lý và phân chia dữ liệu.

Tiền xử lý:

- Cân bằng sáng cho ảnh: Mặc dù các thông số độ sáng và thông số góc của ảnh đã được căn chỉnh bởi máy chụp nhưng trong quá trình chụp tự động không thể tránh khỏi quá trình bị nhiễu, bởi tác động của ngoại vật. Vì vậy ta cần căn chỉnh lại độ sáng để dữ liệu học được chuẩn hơn.

- Cân bằng sáng (Histogram):

- Điều chỉnh histogram về trạng thái cân bằng, giá trị các điểm ảnh không bị co cụm tại một khoảng nhỏ mà được "kéo dãn".



Hình 5.3 Mô tả qua trình histogram hóa.

- Mặc định ảnh màu là hệ RGB hoặc BGR, muốn cân bằng sáng ta cần biến đổi về hệ màu HSV. Hệ màu HSV bao gồm 3 channel:

- H-HUE: giá trị màu
- S-SATURATION: độ bảo hòa.
- V- VALUE: độ sáng của màu sắc.

- Ta sẽ can thiệp vào thông số V để cân bằng sáng cho ảnh màu.

- Điều chỉnh kích thước ảnh

- Trong quá trình chụp bằng máy chụp tự động kích thước của ảnh khó kiểm soát và dễ bị thay đổi bởi ngoại cảnh. Đối với mô hình neural tích chập ta cần dữ liệu kích thước ảnh đầu vào là như nhau để khi tiến hành trích suất bản đồ đặc tính (feature map) các thông số tích chập ít bị sai lệch nhất. Đầu tiên ta sẽ xác định cạnh ngắn hơn của ảnh, từ đó tính ra tọa độ tâm. Từ tọa độ trung tâm này, lấy ra một ma trận vuông có kích thước bằng kích thước cạnh ngắn hơn của ảnh.

Lấy giá trị nhãn của từng ảnh:

- Khi lưu dữ liệu, ảnh được phân vào từng thư mục có tên thư mục chính là số thứ tự phân lớp mà ảnh đó thuộc về. Do đó khi lấy giá trị nhãn của ảnh, nghĩa là lấy thông tin ảnh đó đang thuộc lớp nào, ta sử dụng thông tin của đường dẫn ảnh.

Tạo file train và file test:

- Sau khi bước tiền xử lý được thực hiện, chúng ta sẽ tạo file data để training và test. Ở đây sẽ chia data ra làm 3 phần, 80 % (1874 images) cho training và 20% (469 images) data cho tập testing.

- Ở tập training ta tiếp tục lấy ra 20% (374 images) để cho tập validation.

Sử dụng mô hình pre-trained:

- Ở đây nhóm chúng tôi sử dụng pre-trained của model CSP-darknet53 để lấy file trọng số (weight) mà tác giả đã train trên ImageNet.

5.5) Xây dựng mô hình nhận dạng nhân trắc bằng YOLOv4 sử dụng Google Colab.

- Trong model darknet có nhiều thông số ảnh hưởng tới tốc độ huấn luyện cũng như tính chính xác của hệ thống. Ở đây chúng tôi chỉ chỉnh sửa những thông số quan trọng ảnh hưởng trực tiếp tới những điều nói trên, những thông số khác vẫn giữ nguyên theo setup ban đầu của tác giả.
 - Batch size: Nhóm chúng tôi chọn batch size với kích thước là 64. Các trọng số sẽ được cập nhập với gradient trung bình của từng batch.
 - Iterations: Số lần lặp được nhóm chọn là 50000 lần lặp tương ứng với 50000 lần cập nhập tham số. Tuy nhiên khi hàm Loss trở nên bão hòa, quá trình lặp sẽ được dừng lại.
 - Learning rate decay: Tốc độ học khởi tạo là 0.001 với decay là 0.005
 - Momentum: Momentum được đặt với giá trị 0.949
 - Input size: Kích thước ảnh ngõ vào sẽ được resize lại thành 416 x 416 x 3.
 - Max_batches: Kích thước là 66000 tức là sau khi huấn luyện đủ 66000 batches hoặc model bão hòa thì ta sẽ dừng huấn luyện.

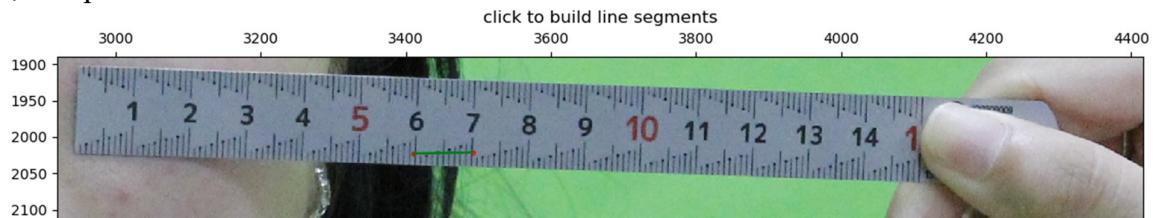
5.6) Tính toán khoảng cách số đo các góc trên khuôn mặt.

- Dựa vào những vị trí đã được xác định ở chương trước ta đã biết được vị trí tọa độ của từng điểm trên khuôn mặt và hai tai, sử dụng những dữ liệu đó để tính toán các góc và khoảng cách các điểm với nhau trên bức hình.
 - Lưu lại các thông số kích thước khoảng cách, góc ra file .csv để có thể trích suất lấy thông tin và tiện lưu trữ.
- 5.6.1) Tính toán hệ số pixel trên hình đầu vào, xác định thông số góc và khoảng cách.**
 - Sử dụng thư viện mathplotlib để đưa ảnh vào hệ trực Descartes với hai trục (x,y) bằng với hai chiều weight và height của ảnh đầu vào.



Hình 5.6.1 : Kết quả nhận diện các điểm nhân trắc trên mặt (hướng chính diện)

- Tiếp theo ta xác định xem, 1cm trên ảnh ứng với bao nhiêu pixel. Để làm được điều đó, ta sẽ chọn 2 điểm ứng với 1cm trên cây thước ở trong hình từ đó nội suy ra được số pixel.



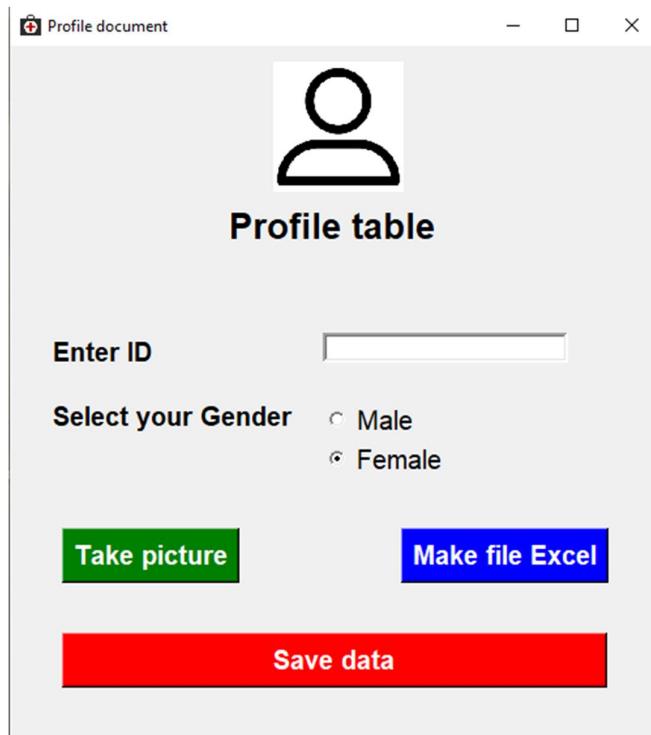
- Hình 5.6.2 : Xác định số pixel ứng với 1 cm trên ảnh.

- Sau khi xác định được số pixel trong ảnh, tiến hành lấy tỉ số của một mm/pixel. Từ đó ta xác định được tỉ lệ pixel của ảnh so với kích thước ngoài đời thực.
- Sau khi đã có được tỉ lệ 1mm/pixel ta nội suy được tọa độ khoảng cách giữa các điểm dựa trên file thu thập nhân trắc ở phần giới thiệu bằng cách lấy tỉ số vừa tính được nhân cho khoảng cách giữa hai điểm theo trục x hoặc theo trục y.
- Để tính góc tạo thành giữa 3 điểm, ta xác định 3 tâm của bounding box có liên quan và nối chúng lại với nhau.
- Có được các thông số khoảng cách và góc giữa các điểm, tiến hành lũ trũ ra file .csv

5.7) Tạo giao diện GUI và kết nối với mySQL.

5.7.1) Giao diện

- Trong thực tiễn, các y bác sĩ hoặc các nhà nghiên cứu muốn lấy được số đo kích thước nhân trắc thường thực hiện các bước sau:
 - ❖ Đo bằng phương pháp lấy mẫu thủ công bằng tay:
 - Bước 1: Đánh số mẫu, người lấy mẫu sử dụng thước kẹp, thước kẻ góc để đo trực tiếp lên khuôn mặt, với 10 lần lấy mẫu.
 - Bước 2: Ghi lại số liệu bằng phiếu thu thập nhân trắc.
 - Bước 3: Lưu trữ lại số liệu vào file excel hoặc trên database
 - ❖ Đo bằng phương pháp gián tiếp (chụp hình mẫu rồi đưa lên phần mềm tính toán):
 - Bước 1: Đánh số mẫu ảnh, chụp mẫu ảnh theo nhiều góc khác nhau.
 - Bước 2: Sử dụng phần mềm tính toán chuyên dụng để lấy số liệu nhân trắc trên người (kéo thả bằng tay).
 - Bước 3: Lưu trữ các số liệu vừa đo được vào file excel hoặc lên database.
 - ❖ Đo bằng phương pháp quét 3D:
 - Bước 1: Đánh số mẫu ảnh, sử dụng máy ảnh 3D quét bộ phận cần tính toán nhân trắc.
 - Bước 2: Sử dụng phần mềm để tính toán các góc trên mô hình 3D vừa quét được.
 - Bước 3: Lưu trữ các số liệu vừa đo được vào file excel hoặc lên database
- Nhận thấy tất cả các bước trên đều có chung một quy tắc là chụp hình người lấy mẫu, đo kích thước nhân trắc và lưu trữ dữ liệu. Do đó nhóm nghiên cứu quyết định xây dựng một giao diện tích hợp đầy đủ tính năng để người lấy mẫu dễ dàng sử dụng mà không tốn quá nhiều thời gian.
- Sử dụng thư viện Tkinter trong python để thiết lập giao diện với mục đích đơn giản hóa quá trình thao tác, giúp người dùng dễ dàng sử dụng ở bất cứ đâu mà không đòi hỏi tính chuyên môn cao.



Hình 5.7.1 Giao diện phần mềm.

- Hàng đầu sẽ yêu cầu mọi người nhập mã ID của folder ảnh, ví dụ như ngày chụp, hay số lần chụp vv
- Hàng thứ 2 là giới tính, yêu cầu người dùng chọn là nam hoặc nữ
- Button “Take picture” có công dụng kết nối với máy ảnh trên kệ, chọn folder truyền ảnh và gửi ảnh chụp được từ máy ảnh vào folder làm việc của phần mềm.
- Nút nhấn “Make file Excel” có công dụng nhận dạng các đặc tính trên khuôn mặt, xuất file tính toán ra đuôi .csv
- Nút nhấn “Save data” sẽ gửi những file tính toán đuôi csv vừa xuất lên database và lưu trữ các file dưới dạng table.

The screenshot shows a database interface with a 'Result Grid' containing 16 rows of data. The columns are labeled 'ID', 'Comment', and 'Parameter'. The data includes various abbreviations like dn_prn, dn_sn, etc., with numerical values. Below the grid is an 'Output' section showing two log entries: a SELECT query returning 9 rows and an INSERT query returning 40 rows.

ID	Comment	Parameter
1	dn_prn	53
2	dn_sn	69
3	dsn_prn	16
4	g_n_prn	147.467
5	n_prn_gn	131.964
6	g_sn_gn	165.164
7	n_pm_sn	95.1815
8	pm_sn_ls	126.287
9	n_8_sn	22.8958
10	sn_8_gn	24.8083
11	prn_n_sn	20.2985
12	dsn_gn	87
13	dch_gn	59
14	RP_GP	24
15	AB	43.7858
16	AC	57.4251

medical 1

Output:

#	Time	Action	Message
1	10:22:00	SELECT * FROM testing.sample LIMIT 0, 1000	9 row(s) returned
2	10:22:47	SUPER INSERT * FROM testing.medical LIMIT 0, 1000	40 rows(s) returned

Hình 5.7.2 Dựa data dữ liệu lên SQL

- Chúng ta có thể xem trực tiếp trên giao diện nhưng bảng table này

The screenshot shows a 'Customer List' application window. It features a main grid displaying data with columns: Customer ID, Comment, and Parameter. The data is identical to the one shown in the previous screenshot. Below the grid are buttons for 'Export CSV', 'Import CSV', 'Save Data', and 'Exit'. At the bottom is a search bar with fields for 'Search' and 'Clear'.

Customer List		
Customer ID	Comment	Parameter
1	dn_prn	53.0
2	dn_sn	69.0
3	dsn_prn	16.0
4	g_n_prn	147.467
5	n_prn_gn	131.964
6	g_sn_gn	165.164
7	n_pm_sn	95.1815
8	pm_sn_ls	126.287
9	n_8_sn	22.8958
10	sn_8_gn	24.8083
11	prn_n_sn	20.2985
12	dsn_gn	87.0
13	dch_gn	59.0
14	RP_GP	24.0
15	AB	43.7858
16	AC	57.4251
17	AB_1	86.122
18	D_AB	24.3956

Export CSV Import CSV Save Data Exit

Search

Search Search Clear

Hình 5.7.3 Lưu dữ liệu vừa tính được lên database

CHƯƠNG 6 : KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ.

Chương này sẽ đưa ra kết quả nghiên cứu, chế tạo máy phát hiện các vùng nhân trắc, độ chính xác nhận dạng các vùng điểm nhân trắc trên khuôn mặt từ đó xuất ra file đo đặc. So sánh các giá trị vừa tính được với giá trị thực tế, đưa ra nhận xét.

6.1) Kết quả thực nghiệm:

STT	Hạng mục đánh giá	Yêu cầu kỹ thuật	Kiểm nghiệm thực tế
1	Độ ổn định	Không	Hoạt động trên 40 lần
2	Góc quay	Góc $0^\circ, 90^\circ, 180^\circ, 270^\circ$	Đạt yêu cầu về góc quay
3	Mức độ nhận dạng	Đủ tất cả các điểm nhận dạng	Chính xác 98%
4	Tốc độ và chất lượng tạo file thu thập nhân trắc	Đủ tất cả các góc và khoảng cách Tốc độ tạo file 30s/file	Đạt yêu cầu



Hình: Máy xử lý nhận dạng nhân trắc tự động.

Mức độ nhận dạng:

- Kết quả ảnh mô phỏng với tập ảnh bắt kì :



Hình 6.1.1 : Kết quả nhận diện các điểm nhân trắc trên mặt (hướng chính diện)



Hình 6.1.2: Kết quả nhận diện các điểm nhân trắc tai và mũi (tai phải)



Hình 6.1.3: Kết quả nhận diện các điểm nhân trắc tai và mũi (tai trái)

A	B	C	D	E	F
1 Comment	Parameter				
2 dn_prn	32.76824				
3 dn_sn	39.68361				
4 dsn_prn	10.3748				
5 g_n_prn	140.9898				
6 n_prn_gn	144.5954				
7 g_sn_gn	175.1783				
8 n_prn_sn	114.7751				
9 prn_sn_ls	119.7689				
10 n_13_sn	22.82902				
11 sn_13_gn	29.32991				
12 prn_n_sn	17.46747				
13 dsn_gn	61.341				
14 dch_gn	34.62589				
15 RP_GP	18.80432				
16 AB	[14.76745527]				
17 AC	[32.22776881]				
18 AB_1	51.96356				
19 D_AB	8.265467				
20 AB_2	101.0853				
21 D_AB2	13.63038				
22 dtr_n	74.69855				
23 d1_2	[19.94192416 49.5246461]				
24					
25					
--					

A	B	C	D	E
1 Comment	Parameter			
2 diem n so Down				
3 diem n so Up				
4 Co diem k None				
5 Chieu cao	42.49383			
6 Chieu dai	28.57992			
7 Do nho doi	13.91391			
8 Chieu ron	36.35165			
9 Chieu dai	34.97279			
10 Chieu dai doan mf				
11				
12				
13				
14				
15				

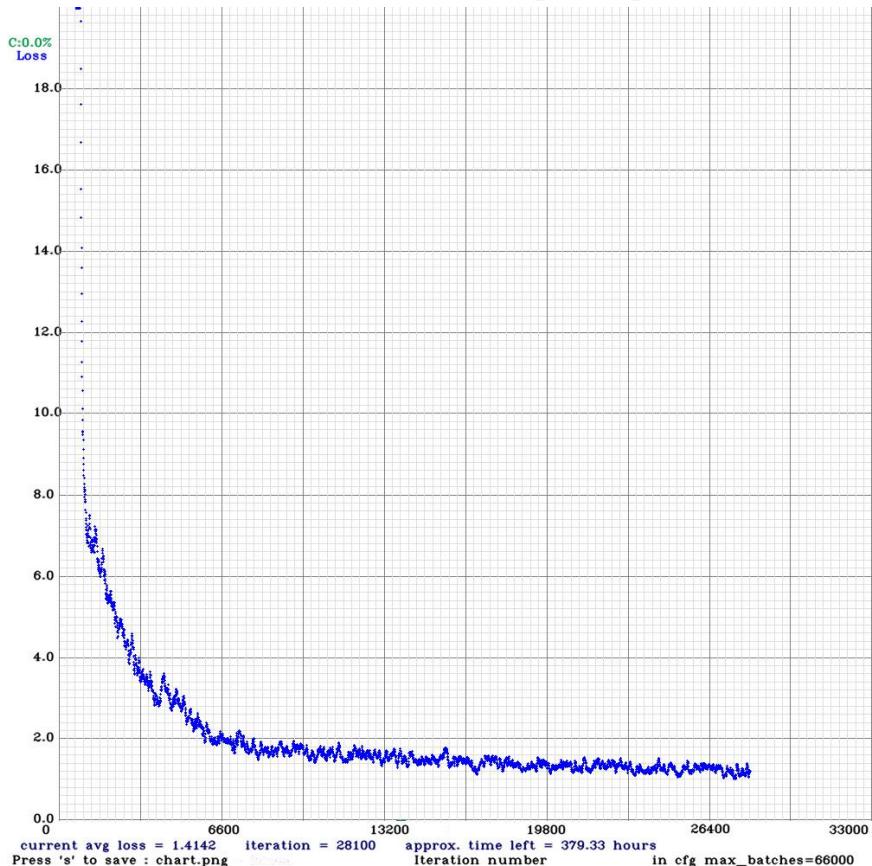
Hình 6.1.4 : Tọa độ góc và khoảng cách được lưu trữ trên file .csv

6.2) Đánh giá thực nghiệm

BẢNG 5.2: ĐỘ CHÍNH XÁC THUẬT TOÁN TRONG QUÁ TRÌNH XÂY DỰNG MODEL.

	Tập huấn luyện (<i>Training</i>)	Tập kiểm định (<i>Validation</i>)	Tập kiểm tra (<i>Testing</i>)
Độ chính xác	98,2258%	99,5326%	97,8693%
Sai số trung bình hệ thống.	1.435%		

- Từ bảng trên có thể kết luận, mô hình đã cho ta thấy kết quả khá tốt với avg lossfunction ở 1,435% với model nhận diện vật thể phức tạp.



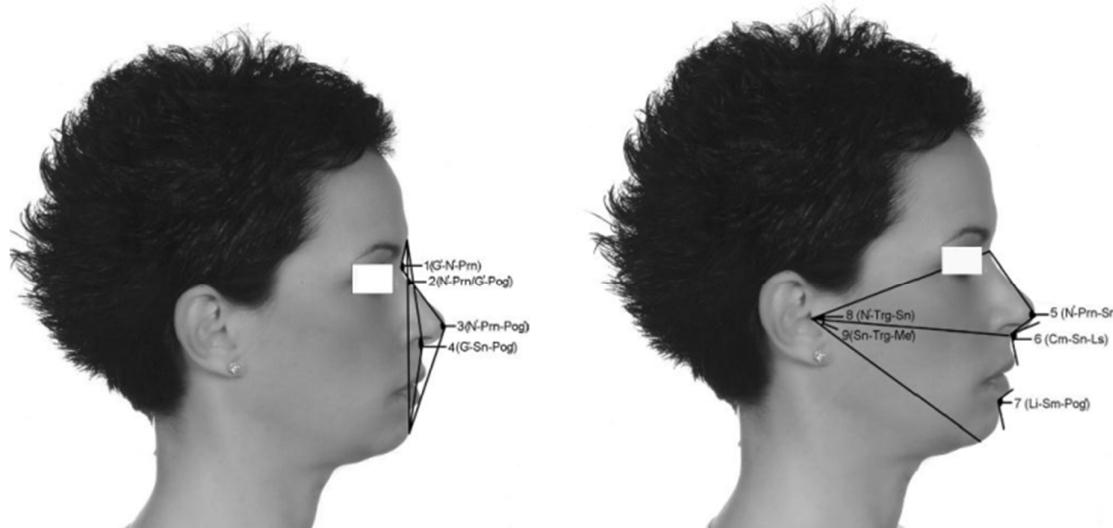
Biểu đồ 6.2.1: Biểu đồ thể hiện số vòng lặp và sai số trung bình của hệ thống.

- Độ chính xác mAP ở tập kiểm đối với hình ảnh của máy chụp tự động, hoặc hình ảnh có phông nền đơn giản với đầy đủ ánh sáng cho hiệu quả khá cao 97%.

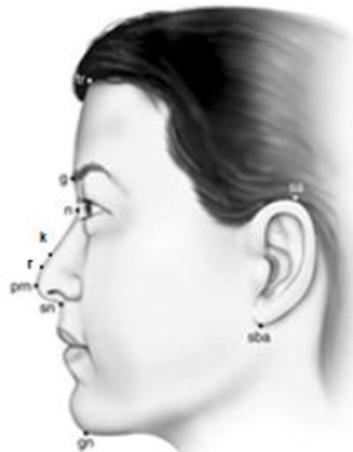
Nhưng khi ảnh đầu vào bị tác động nhiều bởi ngoại cảnh và ánh sáng, độ chính xác của model giảm mạnh giao động từ (40% – 70 %).

- ❖ So sánh các dữ liệu điểm tính toán tự động với dữ liệu tính toán thủ công trên phần mềm.

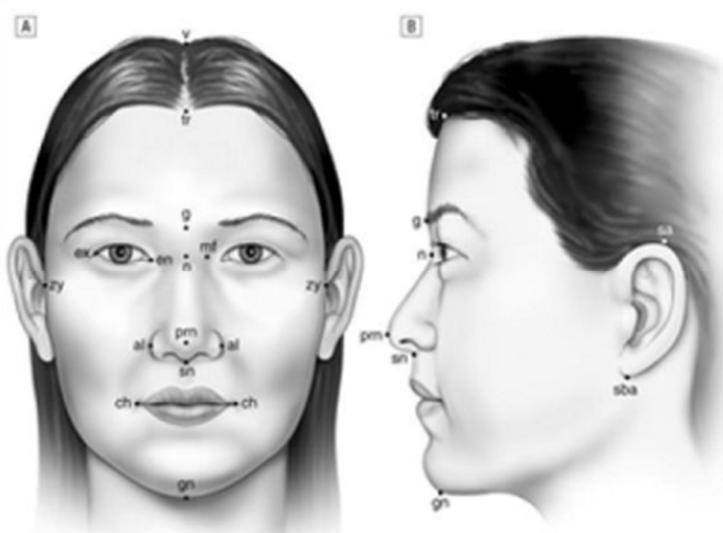
BẢNG 6.2.1: ĐO NHÂN TRẮC QUA ẢNH KỸ THUẬT SỐ

STT	CÂU HỎI	CÂU TRẢ LỜI	
		Tự động	Thực tế
ĐO GÓC			
			
g1	Góc mũi trán (1):g – n – prn	139,98°	139,12°
g2	Góc mũi cằm (3):n – prn – gn	144,59 °	146,14°
g3	Góc (4):g – sn – gn	175,49 °	175,56 °
g4	Góc (5):n – prn – sn	114,78 °	115,24 °
g5	Góc (8):n – 13 – sn	22,83 °	23,79 °
g6	Góc mũi mặt:prn – n – sn	17,47 °	17,50 °
g7	Góc n – k – r (nếu có điểm k)	Không	Không

Góc: n – k – r



ĐO KHOÁNG CÁCH



k1	Vị trí của điểm gốc mũi (n) so với điểm hàm trán (mf)	Dưới	Dưới
k2	Vị trí của điểm gốc mũi (n) so với điểm khói mắt trong (en)	Trên	Trên
k3	Khoảng cách gốc mũi: khoảng cách giữa hai điểm hàm trán (mf)	14,56	15,12

k4	Khoảng cách gian khoe mắt trong: khoảng cách giữa hai điểm khoe mắt trong (en)	34,81	35,01
k5	Chiều dài sống mũi: khoảng cách giữa điểm gốc mũi (n) và điểm đỉnh mũi trên da (prn).	28,44	28,15
k6	Mũi có điểm gù xương (k) hay không?	Không	
k7	Khoảng cách giữa điểm gốc mũi (n) và điểm gù xương (k)		
k8	Khoảng cách giữa điểm gù xương (k) và điểm gù xương và sụn (r)		
k9	Khoảng cách giữ điểm giữa điểm gốc mũi (n) và điểm gù xương (r)		
k10	Độ rộng chóp mũi: khoảng cách của điểm giữa 2 ben cánh mũi (al')		
k11	Chiều dài vành tai: khoảng cách giữa điểm lồi cao nhất ở trên (1) và điểm lồi nhát ở dưới (2) của vành tai	6,13	6,12
k12	Độ nhô đỉnh mũi: Khoảng cách giữa điểm dưới mũi (sn) và điểm đỉnh mũi trên da (prn)	18,84	18,02
k13	Chiều rộng mô mềm mũi (al)	36,18	35,75
k14	Vị trí của điểm gù xương (k) so với trung điểm(1/2) của đoạn n-prn	Không có	
k15	Đo khoảng cách sau: 	a) 73,93 b) 39,38 c) 60,71	a) 74,12 b) 38,78 c) 60,52

		d) 34,27	d) 33,15
k16	<p>Đo khoảng cách sau:</p>	<p>Từ điểm gốc mũi n(C) kẻ vuông góc với đường thẳng ngang (AB) kẻ từ điểm đỉnh mũi prn (B), điểm giao nhau (A) tại rãnh cánh mũi</p>	<p>AC = 18,16 AC = 20,24 AB = 31,90 AB = 31,84</p>
k17	<p>Đo khoảng cách sau:</p>	<p>Khoảng cách từ điểm gốc mũi n (A) đến điểm da môi trên ls (B)</p>	<p>AB = 61,15 AB = 58,89</p>

		Khoảng cách từ điểm đỉnh mũi prn (D) đến hình chiếu vuông góc của nó trên đường AB
	CD = 13,36	CD = 13,55

6.3) Đánh giá độ chính xác của lấy dữ liệu tự động và lấy dữ liệu bằng tay.

Confusion Matrix recognized nasal anthropology											
Model Prediction	tr(Trichion)	2	0	3	0	0	1	0	1	1	-12
	g (Glabella)	-1	8	0	2	2	3	0	0	2	2
	n(Nasion)	-2	1	6	3	1	1	0	4	0	-10
	en(Endocanthion)	-4	0	1	6	1	3	0	0	1	-8
	ex(Exocanthion)	-0	2	1	2	4	2	1	3	1	-6
	prn(Pronasale)	-0	2	3	3	2	6	2	1	1	-4
	r(Rhinion)	-2	1	1	0	2	0	12	0	2	-2
	sn(Subnasale)	-1	1	2	0	2	0	1	9	1	0
	al(Alare)	-1	1	1	1	0	4	4	0	8	0
	pg (Pogonion)	-0	1	0	0	1	1	1	5	1	10
Actual											-0

Hình 6.3.1 Đánh giá độ chính xác của dữ liệu

⇒ Ta thấy kết quả của máy khá gần với kết quả tính toán bằng tay nhưng tốc độ thì nhanh hơn rất nhiều

6.4) Tổng quan đánh giá về thiết bị

❖ Điểm mạnh:

- Tốc độ đo kiểm nhanh, hoạt động ổn định không xảy ra lỗi trong quá trình vận hành.
- Giá thành rẻ, nhận dạng được hầu hết các điểm nhận dạng
- Tính chính xác cao.
- Lưu trữ file lên database và truy suất một cách nhanh chóng.
- Nhỏ gọn, dễ di chuyển, gá đặt, thuận tiện làm việc ở mọi nơi.

❖ Điểm yếu:

- Máy chưa nhận dạng nhân trắc chưa xác định chính xác hoàn toàn tất cả các điểm được đề ra. Bên cạnh đó, cách gá đặt máy ảnh ảnh, background ảnh hướng khá nhiều tới độ chính xác của ảnh nhận dạng.

CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

7.1) Kết luận.

- Nhìn chung nhóm nghiên cứu thiết kế thành công máy và xử lý nhận dạng nhân trắc (tai, mũi). Máy đáp ứng được các yêu cầu kỹ thuật đã được đề ra từ bên đặt hàng: Tốc độ xử lý dữ liệu nhanh (3 phút/1 người), gọn nhẹ dễ mang vác (khối lượng 15kg), giá thành phải chăng và độ chính xác thỏa mãn điều kiện đo đạt được yêu cầu (sai lệch so với dữ liệu thật 3-5%).

7.2) Hướng phát triển

- Tính toán tái cấu trúc khuôn mặt 3D từ các tọa độ đã nhận dạng.
- Xây dựng thuật toán nội suy cấu trúc mũi, tai từ các số liệu và tọa độ điểm đã nhận dạng.
- Ứng dụng trong việc tinh chỉnh tỉ lệ khuôn mặt, tai,..
- Ứng dụng vào khảo sát sự ảnh hưởng của môi trường tác động tới nhân trắc của mỗi người.
- Ứng dụng vào tái cấu trúc các bộ phận bị mất.

PHỤ LỤC 1

Điều chỉnh kích thước ảnh

```
def resize2SquareKeepingAspectRatio(img, size, interpolation):
    h, w = img.shape[:2]
    c = None if len(img.shape) < 3 else img.shape[2]
    if h == w: return cv2.resize(img, (size, size), interpolation)
    if h > w: dif = h
    else: dif = w
    x_pos = int((dif - w)/2.)
    y_pos = int((dif - h)/2.)
    if c is None:
        mask = np.zeros((dif, dif), dtype=img.dtype)
        mask[y_pos:y_pos+h, x_pos:x_pos+w] = img[:h, :w]
    else:
        mask = np.zeros((dif, dif, c), dtype=img.dtype)
        mask[y_pos:y_pos+h, x_pos:x_pos+w, :] = img[:h, :w, :]
    return cv2.resize(mask, (size, size), interpolation)
```

Điều chỉnh độ sáng

```
img_yuv = cv2.cvtColor (img, cv2.COLOR_BGR2YUV)
img_yuv[:, :, 0] = cv2.equalizeHist (img_yuv[:, :, 0])

# convert the YUV image back to RGB format
img_output = cv2.cvtColor (img_yuv, cv2.COLOR_YUV2BGR)
```

Lấy giá trị label của thư mục ảnh

```
def get_class(img_path):
    return int(img_path.split('/')[-2])
```

(Với *img_path* là đường dẫn tới ảnh.)

Tạo file train và file test.

```
files = []
for ext in ["*.png", "*.jpeg", "*jpg"]:
    image_files = glob2.glob(os.path.join("data/data/", ext))
    files += image_files

nb_val = math.floor(len(files)*0.2)
rand_idx = np.random.randint(0, len(files), nb_val)

# Tạo file train.txt
with open("train.txt", "w") as f:
    for idx in np.arange(len(files)):
        if (os.path.exists(files[idx][:-3] + "txt")):
            f.write(files[idx]+'\n')

# Tạo file vali.txt
with open("val.txt", "w") as f:
    for idx in np.arange(len(files)):
        if (idx in rand_idx) and (os.path.exists(files[idx][:-3] + "txt")):
            f.write(files[idx]+'\n')
```

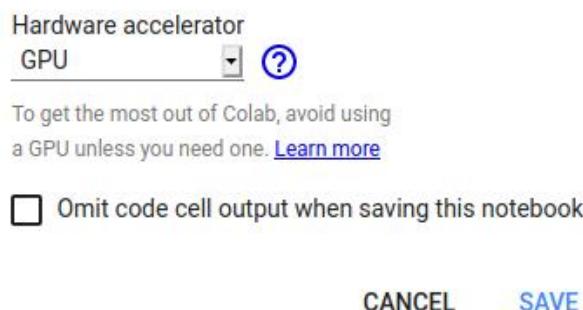
❖ Google Colab

- Google Colab hay Colaboratory là sản phẩm của Google Research cho phép chạy các dòng code python qua trình duyệt nhằm phục vụ các nhu cầu về giáo dục và nghiên cứu. Google Colab cũng không yêu cầu cài đặt cấu hình cho máy tính mà các tài nguyên CPUs, GPUs, TPUs đều được cung cấp hoàn toàn miễn phí. Tuy nhiên vì tính miễn phí, Google Colab giới hạn thời gian sử dụng trong vòng 12 tiếng.
 - Trên môi trường Google Colab cũng có cài đặt sẵn một số thư viện dành cho Deep Learning phổ biến như PyTorch, TensorFlow, Keras, OpenCV... Ngoài ra, ưu điểm của Google Colab là có thể kết nối với Google Drive để đọc và lưu dữ liệu nên rất tiện dụng.

❖ Huấn luyện dữ liệu

- Ta truy cập vào Google Colab theo đường dẫn url sau:
"https://colab.research.google.com" và tạo một new notebook mới để chuẩn bị huấn luyện dữ liệu.
- Thực hiện thay đổi runtime type từ CPU sang GPU.

Notebook settings



Hình 5.6.5 : Thay đổi runtime type của Google Colab để sử dụng GPU

- Tiếp theo để sử dụng các tập tin được tải lên từ Google Drive ta thực hiện liên kết Google Colab với Google Drive bằng dòng lệnh sau:

The screenshot shows a terminal window in Google Colab. The user has run the command `from google.colab import drive
drive.mount('/content/gdrive')`. A message from Google Drive asks the user to go to a URL in a browser: <https://accounts.google.com/o/oauth2/auth>. Below the URL, it says 'Enter your authorization code:' followed by a series of dots. At the bottom, it shows 'Mounted at /content/gdrive'.

Hình 5.6.6 : Liên kết Google Drive với Google Colab

- Tải mã nguồn darknet của YOLO về drive để tạo 1 thư mục backup giúp lưu cái trọng số sau mỗi lần lặp giúp tránh bị kill sau 12 giờ. Nếu sau 12 giờ bị Google Colab kill toàn bộ quá trình huấn luyện, chúng ta chỉ cần lấy lại file trọng số đã được lưu sẵn trong folder backup để tiếp tục huấn luyện.

```

# Step 2. Tai ma nguon YOLO ve drive
!rm -rf darknet
%cd /content/gdrive/My\ Drive
!git clone https://github.com/AlexeyAB/darknet

%cd /content/gdrive/My\ Drive/darknet
!rm -rf data
!mkdir data

```

- Sau khi tải mã nguồn darknet, Google Drive sẽ xuất hiện thư mục darknet. Trong thư mục data của darknet, xóa data có sẵn và đưa data nhận dạng cùng file txt đã gán nhãn. Trong thư mục darknet, ta truy cập vào Makefile thay đổi GPU = 1 và CUDNN=1 để sử dụng GPU từ Google Colab, đổi OPENCV = 1 để sử dụng thư viện opencv. Với file "yolov4-custom.cfg" trong thư mục .cfg thay đổi các tham số và số lớp như đã đề cập ở phần trên.

- Tạo file data chứa các dữ liệu các tập tin để huấn luyện gồm:
 - 2 file train (train.txt) và validation (vali.txt) chia ngẫu nhiên các tập dữ liệu vào 2 file này
 - 1 file yolo.names chứa tên các nhãn số classes cần huấn luyện
 - 1 file backup
 - Biên dịch mã nguồn của darknet, tải pretrain weight của tác giả và tiến hành huấn luyện

```

# Step 8. Make darknet ( Biên dịch mã nguồn)
%cd /content/gdrive/My\ Drive/darknet
!rm darknet
!make

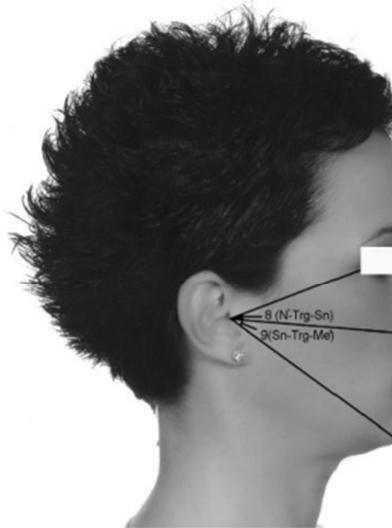
```

```

# Step 9. Download pretrain weight ( Tải file pre train)
%cd /content/gdrive/My\ Drive/darknet
!wget
https://github.com/AlexeyAB/darknet/releases/download/darknet\_yolo\_v3\_optimal/yolov4.conv.137

```

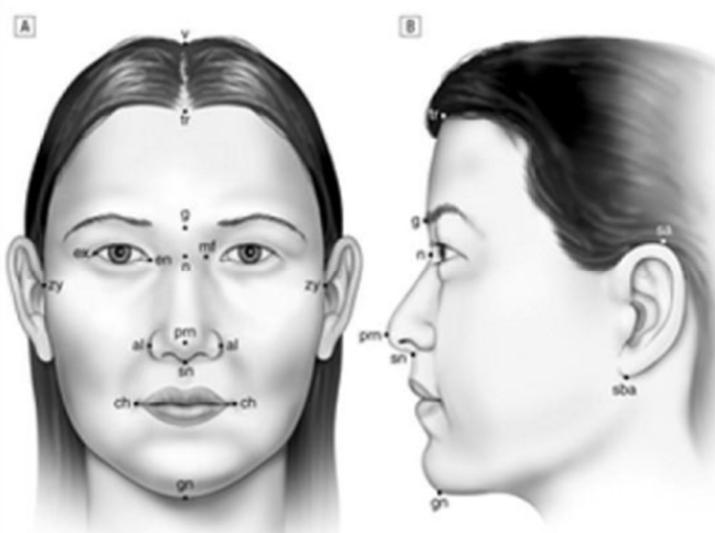
PHỤ LỤC 2
BẢNG 1.1: ĐO NHÂN TRẮC QUA ẢNH KỸ THUẬT SỐ

STT	CÂU HỎI	CÂU TRẢ LỜI
ĐO GÓC		
	 <p>1 (G-N-Prn) 2 (N-Prn/G-Pog) 3 (N-Prn-Gn) 4 (G-Sn-Pog)</p>	 <p>5 (N-Prn-Sn) 6 (Cm-Sn-Ls) 7 (Li-Sm-Pog) 8 (N-Trg-Sn) 9 (Sn-Trg-Me)</p>
g1	Góc mũi trán (1):g – n – prn	
g2	Góc mũi cằm (3):g – prn – gn	
g3	Góc (4):g – sn – gn	
g4	Góc (5):g – prn – sn	
g5	Góc (8):n – tr – sn	
g6	Góc mũi mặt:prn – n – sn	
g7	Góc n – k – r (nếu có điểm k)	

Góc: n – k – r



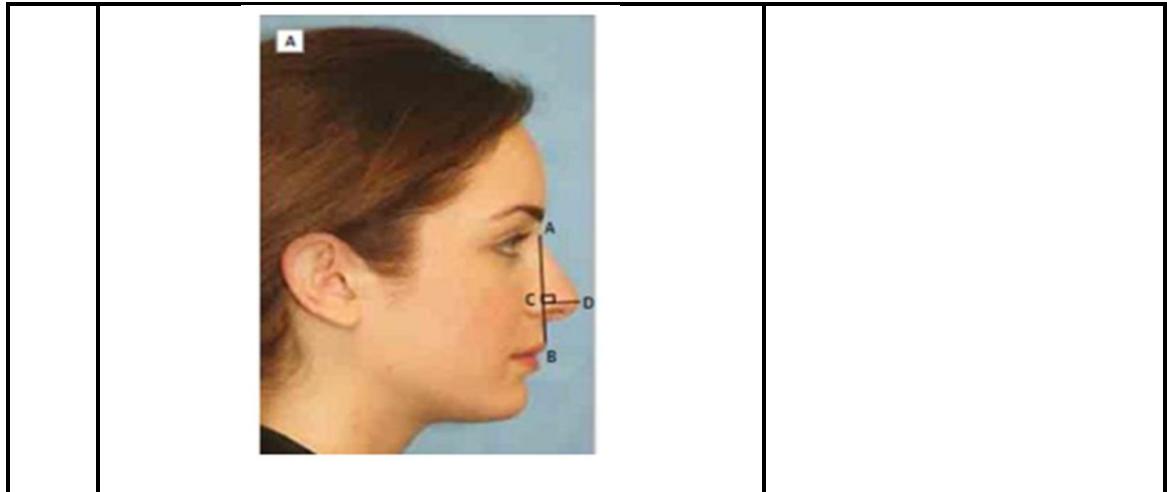
ĐO KHOẢNG CÁCH



k1	Vị trí của điểm gốc mũi (n) so với điểm hàm trán (mf)	1. Trên 2. Dưới
k2	Vị trí của điểm gốc mũi (n) so với điểm khóe mắt trong (en)	1. Trên 2. Dưới
k3	Khoảng cách gốc mũi: khoảng cách giữa hai điểm hàm trán (mf)	$mf - mf$

k4	Khoảng cách gian khoe mắt trong: khoảng cách giữa hai điểm khoe mắt trong (en)	en – en
k5	Chiều dài sống mũi: khoảng cách giữa điểm gốc mũi (n) và điểm đỉnh mũi trên da (prn).	n – prn
k6	Mũi có điểm gù xương (k) hay không?	1. Có 2. Không -> chuyển tới k10
k7	Khoảng cách giữa điểm gốc mũi (n) và điểm gù xương (k)	
k8	Khoảng cách giữa điểm gù xương (k) và điểm giữa xương và sụn (r)	
k9	Khoảng cách giữ điểm giữa điểm gốc mũi (n) và điểm gù xương (r)	
k10	Độ rộng chót mũi: khoảng cách của điểm giữa 2 ben cánh mũi (al')	
k11	Chiều dài vành tai: khoảng cách giữa điểm lồi cao nhất ở trên (1) và điểm lồi nhát ở dưới (2) của vành tai	(1) – (2) =
k12	Độ nhô đỉnh mũi: Khoảng cách giữa điểm dưới mũi (sn) và điểm đỉnh mũi trên da (prn)	sn – prn =
k13	Chiều rộng mô mềm mũi (al)	al – al =
k14	Vị trí của điểm gù xương (k) so với trung điểm(1/2) của đoạn n-prn	1. Trên 2. Dưới
k15	Đo khoảng cách sau: 	a. tr – n b. n – sn c. sn – gn

		d. ch - gn
k16	<p>Đo khoảng cách sau:</p>	<p>Từ điểm gốc mũi n(C) kẻ vuông góc với đường thẳng ngang (AB) kẻ từ điểm đỉnh mũi prn (B), điểm giao nhau (A) tại rãnh cánh mũi</p> <p>AC = AB =</p>
k17	<p>Đo khoảng cách sau:</p>	<p>Khoảng cách từ điểm gốc mũi n (A) đến điểm da môi trên ls (B)</p> <p>AB =</p> <p>Khoảng cách từ điểm đỉnh mũi prn (D) đến hình chiếu vuông góc của nó trên đường AB CD =</p>



DANH MỤC TÀI LIỆU THAM KHẢO

---Ψ---

- [1] Trị Chát- Lê Văn Uyễn, Tính toán thiết kế hệ dẫn động cơ khí (tập 1), NXB Giáo dục
- [2] Trị Chát- Lê Văn Uyễn, Tính toán thiết kế hệ dẫn động cơ khí (tập 2), NXB Giáo dục.
- [3] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. International journal of computer vision, 2010
- [4] 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, 2016
- [5] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. Dataset available from <https://github.com/openimages>, 2017
- [6] <http://www.arduino.vn>
- [7] <http://ainoodle.tech/2019/08/25/yolo-series-train-va-dung-yolo-model-tren-windows/>
- [8] <https://github.com/AlexeyAB/darknet>
- [9] <https://nshopvn.com/>
- [10] <https://www.miai.vn/2020/05/25/yolo-series-train-yolo-v4-train-tren-colab-chi-tiet-va-day-du-a-z/>
- [11] https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_s%C3%A2u
- [12] <https://topdev.vn/blog/su-khac-nhau-giua-ai-machine-learning-va-deep-learning/>
- [13] <https://dothanhblog.wordpress.com/2020/05/08/yolov4/>
- [14] Ngôn ngữ lập trình trên Arduino - Hướng dẫn – Cộng đồng Arduino VN

<http://arduino.vn/reference>

[15] Hướng dẫn điều khiển động cơ servo

<https://vandieukhien.vn/servo-motor-la-gi>