

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG ĐIỆN-ĐIỆN TỬ

LUẬN VĂN THẠC SĨ
**Phân loại tế bào máu dùng mạng neural tích
chập (CNN) kết hợp với mô hình Yolo 5**
NGUYỄN ĐĂNG TIỆP
Tiep.ND211106M@sis.hust.edu.vn
Ngành Kỹ thuật y sinh (KH)

Giảng viên hướng dẫn: TS. Dương Trọng Lượng

Chữ ký của GVHD

Viện: Điện Tử - Viễn Thông

HÀ NỘI, 03/2023

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

BẢN XÁC NHẬN CHỈNH SỬA LUẬN VĂN THẠC SĨ

Họ và tên tác giả luận văn:

Đề tài luận văn:

Chuyên ngành:

Mã số SV:

Tác giả, Người hướng dẫn khoa học và Hội đồng chấm luận văn xác nhận tác giả đã sửa chữa, bổ sung luận văn theo biên bản họp Hội đồng ngày..... với các nội dung sau:

.....
.....
.....
.....
.....
.....
.....

Ngày tháng năm

Giáo viên hướng dẫn

Tác giả luận văn

CHỦ TỊCH HỘI ĐỒNG

Lời cảm ơn

Em xin chân thành cảm ơn các thầy cô Đại học Bách Khoa Hà Nội, đặc biệt các thầy cô bộ môn Kỹ thuật y sinh, đã tận tình dạy dỗ, giúp đỡ và tạo mọi điều kiện tốt nhất cho em trong suốt quãng thời gian em theo học tại trường, để em có thể hoàn thành được đề tài này.

Em tỏ lòng biết ơn sâu sắc với **TS. DUƠNG TRỌNG LƯỢNG**, người thầy đã tận tình hướng dẫn khoa học và giúp đỡ, chỉ bảo em trong suốt quá trình nghiên cứu và hoàn thành luận văn này.

Tôi xin chân thành cảm ơn các bạn học viên cao học khóa 2021A tại Trường Điện – Điện Tử, Đại học Bách Khoa Hà Nội đã giúp đỡ tôi trong quá trình theo học tại trường, cũng như giúp đỡ tôi thực hiện đề tài này.

Xin trân trọng cảm ơn!

Tóm tắt nội dung luận văn

Để chẩn đoán các bệnh lý liên quan đến rối loạn tế bào máu cũng như sức khỏe của con người, một trong những phương pháp quan trọng được áp dụng trong các bệnh viện là việc phân loại và kiểm tra các mẫu máu dưới kính hiển vi. Tuy nhiên, việc kiểm tra các mẫu phẩm máu bằng phương pháp thủ công này gặp nhiều khó khăn khi cần kiểm tra số lượng mẫu phẩm máu lớn và đòi hỏi sự tỉ mỉ, độ chính xác cao. Vì vậy, tác giả lựa chọn đề tài “Phân loại tế bào máu dùng mạng neural tích chập (CNN) kết hợp với mô hình Yolo 5” với một số nội dung cơ bản là : Tìm hiểu về tế bào máu, các bệnh rối loạn máu và phương pháp chẩn đoán bệnh rối loạn máu, nghiên cứu mô hình YOLOv5, mô hình mạng VGG16 kết hợp với YOLOv5, so sánh hai mô hình trong phân loại tế bào máu từ đó xây dựng phần mềm ứng dụng thực tế

Cấu trúc luận văn bao gồm 3 chương :

Chương 1: Cơ sở lý thuyết

Trong chương này đã trình bày về các kiến thức cơ bản liên quan đến máu và các thành phần trong máu, nhấn mạnh vào thành phần tế bào, bao gồm hồng cầu, tiểu cầu và bạch cầu, các bệnh lý máu phổ biến và vai trò của việc phân loại tế bào máu trong việc chẩn đoán và điều trị các bệnh lý này.

Chương 2: Xây dựng mô hình huấn luyện phát hiện phân loại tế bào máu

Trong chương 2, tác giả trình bày về lý thuyết của mô hình YOLOv5 và mô hình YOLOv5 với backbone VGG16 bao gồm cấu tạo, chức năng, ưu nhược điểm của mô hình.

Chương 3 : Kết quả và bàn luận

Trong chương 3, tác giả trình bày quá trình huấn luyện mô hình, các tiêu chí đánh giá một mô hình. Đánh giá và so sánh kết quả huấn luyện mô hình YOLOv5 và mô hình YOLOv5 với backbone VGG16. Từ đó chọn ra mô hình tốt nhất với thông số model tốt để xây dựng phần mềm phân loại tế bào máu. Tác giả đã sử dụng ngôn ngữ lập trình python và công cụ flask API để đưa giao diện phần mềm lên website.

HỌC VIÊN

Ký và ghi rõ họ tên

MỤC LỤC

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT.....	1
1.1 Máu và các thành phần trong máu	1
1.1.1 Thành phần tế bào	1
1.1.2 Thành phần huyết tương	3
1.2 Các bệnh lý máu.....	4
1.2.1 Bệnh lý liên quan đến rối loạn hồng cầu.....	4
1.2.2 Bệnh lý liên quan đến rối loạn tiểu cầu.....	5
1.2.3 Bệnh lý liên quan đến rối loạn bạch cầu	6
1.3 Mục đích và ý nghĩa của việc phân loại tế bào máu	8
1.4 Các phương pháp phân loại tế bào máu đang được sử dụng trong bệnh viện 9	
1.4.1 Phương pháp tế bào học dòng chảy	9
1.4.2 Phương pháp soi mẫu phẩm dưới kính hiển vi	11
1.4.3 Một số phương pháp phân loại tế bào máu tự động.....	13
1.5 Kết luận chương	17
CHƯƠNG 2. XÂY DỰNG MÔ HÌNH HUẤN LUYỆN PHÁT HIỆN, PHÂN LOẠI TẾ BÀO MÁU	18
2.1 Phân loại tế bào bạch cầu sử dụng mô hình mạng YOLOv5.....	18
2.1.1 Sơ lược về mô hình YOLO	18
2.1.2 Mạng YOLOv5	30
2.2 Phân loại tế bào bạch cầu sử dụng mô hình kết hợp VGG16 và YOLOv5 35	
2.2.1 Mô hình mạng VGG16	35
2.2.2 Mô hình kết hợp VGG16 và YOLOv5	39
2.3 Kết luận chương	40
CHƯƠNG 3. KẾT QUẢ VÀ BÀN LUẬN	41
3.1 Cơ sở dữ liệu huấn luyện	41
3.1.1 Mô tả dữ liệu	41
3.1.2 Tăng cường dữ liệu	42
3.1.3 Dữ liệu huấn luyện.....	44
3.2 Công cụ lập trình.....	44
3.3 Chỉ số đánh giá mô hình.....	45
3.3.1 Intersection over Union.....	45

3.3.2	Precision và Recall.....	46
3.3.3	Mean Average Precision	47
3.4	Quá trình huấn luyện các mô hình	49
3.5	Đánh giá hiệu quả của từng mô hình	51
3.5.1	Kết quả thử nghiệm mô hình YOLOv5	51
3.5.2	Kết quả thử nghiệm mô hình VGG16 kết hợp YOLOv5.....	56
3.6	So sánh hai mô hình	58
3.7	So sánh với các mô hình khác được công khai trong các bài báo	58
3.8	Phần mềm phân loại tế bào máu	58
3.8.1	Xây dựng phần mềm	59
3.8.2	Cách sử dụng phần mềm.....	60
3.9	Kết luận chương	62
	KẾT LUẬN	63
	TÀI LIỆU THAM KHẢO	66

DANH MỤC HÌNH VẼ

Hình 1.1 Tỷ lệ của tế bào máu và huyết tương trong máu.....	2
Hình 1.2 Các thành phần của tế bào máu.....	3
Hình 1.3 Nguyên tắc đếm tế bào dòng chảy	11
Hình 1.4 Quy trình thực hiện phết máu ngoại vi	12
Hình 1.5 Mối quan hệ giữa AI, ML, DL	14
Hình 2.1 Kiến trúc mạng của YOLOv1	19
Hình 2.2: Kiến trúc mạng của YOLOv2	20
Hình 2.3: Kiến trúc của YOLOv2 với một skip-connection.....	21
Hình 2.4: Khi có công thức giới hạn tâm của Bounding Box (màu xanh) sẽ nằm trong grid cell sinh ra Anchor Box và ngược lại tâm của Bounding Box (màu trắng) nằm ngoài grid cell sinh ra Anchor Box	22
Hình 2.5: Kiến trúc mạng backbone của YOLOv3.....	23
Hình 2.6: Kiến trúc của YOLOv3 với Feature Pyramid Network (FPN)	24
Hình 2.7: Sự khác nhau giữa Residual Block (bên trái) và CSPResBlock (bên phải)	25
Hình 2.8: So sánh DarkNet53 với CSPDarkNet53	26
Hình 2.9: Chi tiết cụ thể của CSPResBlock.....	26
Hình 2.10: Ở giữa là DropOut, bên phải là DropBlock. Việc loại bỏ đi một nhóm các vị trí gần nhau mang lại nhiều ảnh hưởng đến lớp Convolution hơn	27
Hình 2.11: SPP ban đầu	27
Hình 2.12: SPP trong YOLOv4. Feature map $C \times H \times W$ từ backbone được thực hiện max pooling 4 lần với các kernel có kích thước (1, 3, 9, 13), tạo ra các feature map cùng kích thước với backbone $C \times H \times W$. Các feature map này được concatenate lại với nhau thành feature map $(4 \times C) \times H \times W$	28
Hình 2.13: Ví dụ SPP trong YOLOv4 với $C=512$	29
Hình 2.14: Mô hình mạng PAN	30
Hình 2.15: Sự khác biệt giữa CSPResBlock trong YOLOv4 (trên) và C3 Module (dưới).....	32
Hình 2.16: SPPF của YOLOv5	33
Hình 2.17 Hình ảnh so sánh các mô hình YOLOv5	35
Hình 2.18: Kiến trúc của VGG16	36
Hình 2.19: Bản đồ kiến trúc VGG16	37
Hình 2.20: Kiến trúc mô hình YOLOv5	39
Hình 2.21: Kiến trúc mô hình YOLOv5 backbone VGG16	40
Hình 3.1 Ảnh gán nhãn các tế bào lần lượt theo thứ tự là basophil, diseased, eosinophil, lymphocyte, monocyte, neutrophil	42
Hình 3.2 Sơ đồ chia dữ liệu huấn luyện mô hình.....	44

Hình 3.3 Hình ảnh về công thức tính precision và recall.....	47
Hình 3.4 Biểu đồ heatmap dự đoán các lớp của mô hình yolov5	51
Hình 3.5 Đồ thị F1-confidence.....	53
Hình 3.6 Đồ thị Precision-Confidence	53
Hình 3.7 Đồ thị Recall - Confidence.....	54
Hình 3.8 Đồ thị Precision - Recall	54
Hình 3.9 Hình ảnh huấn luyện mô hình	55
Hình 3.10 Hình ảnh valid mô hình.....	56
Hình 3.11 Đồ thị heatmap dự đoán các lớp của mô hình YOLOv5 backbone VGG16	57
Hình 3.12 Sơ đồ giải thuật	59
Hình 3.13 Giao diện phần mềm	60
Hình 3.14 Kiểm thử phần mềm.....	61
Hình 3.15 Kiểm thử phần mềm.....	61
Hình 3.16 Kiểm thử phần mềm.....	62

DANH MỤC BẢNG BIỂU

Bảng 1.1 Nghiên cứu phân loại tế bào bạch cầu theo mô hình two stage detector	16
Bảng 1.2 Các nghiên cứu phân loại tế bào bạch cầu theo mô hình one stage detector	17
Bảng 3.1 Đánh giá độ hiệu quả mô hình mạng YOLOv5	52
Bảng 3.2 Đánh giá độ hiệu quả mô hình mạng YOLOv5 với backbone VGG16	57
Bảng 3.3 So sánh hai mô hình.....	58
Bảng 3.4 So sánh phương pháp đề xuất với các nghiên cứu khác	58

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Thuật ngữ tiếng anh	Thuật ngữ tiếng việt
1	CNN	Convolutional neural network	Mạng nơ ron tích chập
2	BS		Bác sĩ
3	KTV		Kỹ thuật viên
4	Yolo	You only look one	
5	OD	Object detection	Phát hiện vật thể
6	NMS	Non-maximum Suppression	
7	MRD	Minimal Residual Disease	
8	IoU	Intersection over Union	
9	P	Precision	
10	R	Recall	
11	mAP	Mean Average Precision	
12	AI	Artificial Intelligence	Trí tuệ nhân tạo
13	ML	Machine learning	Học máy
14	DL	Deep learning	Học sâu
15	ANN	Artificial Neural Networks	Mạng nơ ron nhân tạo
16	DNN	Deep Neural Networks	
17	RNN	Recurrent Neural Networks	
18	LSRM	Long Short-Term Memory	
19	SSD	Single Shot MultiBox Detector	
20	CSPNet	Cross Stage Hierarchical Networks	
21	SPP	Spatial Pyramid Pooling	
22	PAN	Path Aggregation Network	
23	VGG	Visual Geometry Group	
24	FPN	Feature Pyramid Network	

I. Đặt vấn đề

Ngày nay với tỉ lệ người mắc bệnh ung thư cụ thể là ung thư máu càng ngày càng tăng, các bác sĩ sẽ phải chỉ định những xét nghiệm thường quy như tổng phân tích tế bào máu, huyết đồ... Sau khi có kết quả xét nghiệm nếu có bất thường các bác sĩ, kỹ thuật y sẽ nhìn hình thái tế bào bằng mắt thường trên kính hiển vi và dựa vào kiến thức, kinh nghiệm có được đưa ra các chẩn đoán về bệnh máu. Việc quan sát dựa vào mắt thường và yêu cầu kiến thức vững chắc và nhiều kinh nghiệm để đưa ra kết luận đó sẽ dẫn đến mất nhiều thời gian; bên cạnh đó các bệnh viện tuyến dưới sẽ không có đủ nguồn nhân lực và kiến thức để thực hiện do đó để có các kết quả chính xác người bệnh phải chuyển tuyến lên tuyến trên hoặc các mẫu bệnh phẩm phải gửi lên các bệnh viện tuyến trên để đọc gây mất thời gian và bỏ lỡ thời gian điều trị tốt nhất cho bệnh nhân.

Do đó tác giả lựa chọn đề tài: “Phân loại tế bào máu dùng mạng neural tích chập (CNN) kết hợp với mô hình Yolo 5” để tìm hiểu, nghiên cứu mô hình phân loại tế bào máu tự động dựa trên dữ liệu tại viện Huyết Học Truyền Máu TW, dữ liệu tại Bệnh Viện Đại Học Y Hà Nội và dữ liệu trên mạng internet. Từ đó, tác giả sẽ chọn ra mô hình tốt nhất xây dựng phần mềm phân loại tế bào máu nhằm áp dụng trong các cơ sở y tế.

II. Mục đích nghiên cứu, đối tượng và phạm vi đề tài

Tìm hiểu một số phương pháp cũng như thuật toán/mô hình phân loại tế bào máu có độ chính xác cao hiện nay. Từ đó xây dựng chương trình phần mềm phân loại tế bào máu, chương trình phần mềm này có thể được ứng dụng trong cơ sở y tế có chuyên khoa huyết học.

Đối tượng nghiên cứu trong đề tài là các loại tế bào bạch cầu bao gồm bạch cầu trung tính, bạch cầu lympho, bạch cầu ưa axit, bạch cầu mono, bạch cầu bazơ và tế bào bạch cầu bất thường

Phạm vi nghiên cứu của đề tài: Đề tài được nghiên cứu trong các khoa huyết học của các bệnh viện và cơ sở y tế trong nước.

III. Ý nghĩa của đề tài

Đề tài đã tìm hiểu về các bệnh lý về máu và phương pháp chẩn đoán các bệnh lý về máu để góp phần làm nổi bật tầm quan trọng của việc phân loại, kiểm tra tế bào

máu, đồng thời tìm hiểu về cách phân tích những thông tin quan trọng từ mẫu phẩm máu để chẩn đoán tình trạng bệnh lý.

Đề tài đã áp dụng kỹ thuật khoa học máy tính vào trong lĩnh vực y tế với việc áp dụng các mô hình học sâu để phân tích và phân loại các hình ảnh tế máu. Các mô hình mạng CNN và mạng YOLOv5 đã được nghiên cứu vào việc xây dựng hệ thống tự động phân loại tế bào máu. Các phương pháp đã đạt độ chính xác cao, thời gian xử lý nhanh. Qua đó cho thấy sự hiệu quả của mô hình học sâu được áp dụng trong phân tích hình ảnh tế bào máu nói riêng và hình ảnh trong y học nói chung.

Việc xây dựng phần mềm tự động phân loại tế bào máu giúp việc chẩn đoán các mẫu phẩm máu dưới kính hiển vi đạt hiệu quả cao, tiết kiệm thời gian phân tích mẫu phẩm, hạn chế những khó khăn khi thực hiện kiểm tra một số lượng lớn mẫu phẩm máu dưới kính hiển vi bằng phương pháp thủ công, hạn chế những khó khăn về điều kiện cơ sở vật chất, kỹ thuật của các BS, KTV tại các cơ sở khám chữa bệnh tuyến dưới

IV. Nội dung nghiên cứu

Trong đề tài tác giả nghiên cứu các nội dung sau:

- Cơ sở lý thuyết về các thành phần máu, các bệnh lý về máu và các phương pháp phân loại tế bào máu.
- Nghiên cứu xây dựng mô hình YOLOv5 và mô hình YOLOv5 với backbone VGG
- Nghiên cứu xây dựng phần mềm phân loại tế bào máu

V. Kết quả đạt được

Đánh giá được mô hình nào tốt hơn giữa YOLOv5 và mô hình YOLOv5 với backbone VGG. Từ thông số tốt nhất của mô hình đó xây dựng phần mềm nhận diện, phân loại tế bào máu.

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

1.1 Máu và các thành phần trong máu

Máu là tổ chức lỏng, lưu thông trong hệ thống tuần hoàn của cơ thể. Máu gồm nhiều thành phần với các chức năng khác nhau và liên quan mật thiết đến chức năng sống của cơ thể. Máu gồm hai thành phần chính: Các tế bào máu và huyết tương [2].

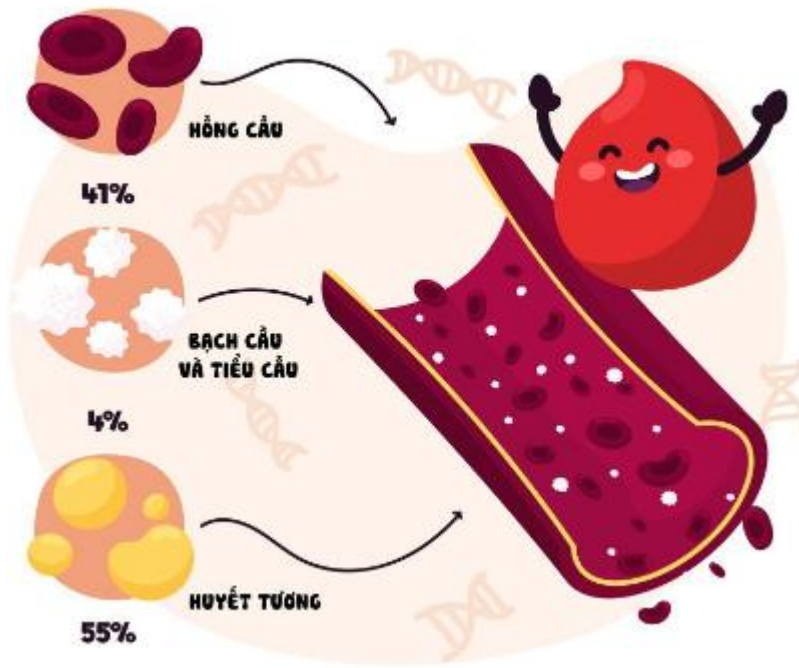
1.1.1 Thành phần tế bào

Tế bào là một đơn vị cấu trúc cơ bản có chức năng sinh học của sinh vật sống. Tế bào là đơn vị nhỏ nhất của sự sống có khả năng phân chia độc lập. Trong cơ thể người có nhiều loại tế bào khác nhau về hình dạng, kích thước và chức năng. Mặc dù khác nhau về nhiều mặt nhưng hầu hết các loại tế bào cũng có 3 phần cơ bản: màng sinh chất, chất tế bào và nhân [2].

Màng sinh chất: Là lớp ngoài của tế bào đặc lại, được cấu tạo từ prô-tê-in và li-pit, có nhiệm vụ thực hiện trao đổi chất với môi trường quanh tế bào [2]. Chất tế bào: Là thành phần nằm trong màng tế bào, gồm nhiều bào quan và chất phức tạp, là nơi diễn ra những hoạt động sống của tế bào. Các bào quan chính là lưới nội chất, ti thể, ri-bô-xôm, bộ máy Gôn-gi, trung thể.

Nhân tế bào: là một bào quan được bao bọc bởi màng tế bào tồn tại bên trong các tế bào nhân thực. Nhân tế bào chứa hết bộ gen của sinh vật, ngoại trừ DNA ty thể, được cuộn lại thành nhiều chuỗi DNA bao gồm các phức hợp protein. Phức hợp protein có nhiều loại protein khác nhau để cấu thành các nhiễm sắc thể. Gen trong các nhiễm sắc thể có cấu trúc đặc thù để thúc đẩy chức năng của tế bào. Nhân tế bào bảo quản độ ổn định của gen và quản lý các hoạt động của tế bào bằng cách điều chỉnh biểu hiện gen.

Do đó, nhân tế bào được coi là trung tâm điều khiển của tế bào [2].



Hình 1.1 Tỷ lệ của tế bào máu và huyết tương trong máu

1.1.1.1. Hồng cầu

Hồng cầu là tế bào không có nhân, chiếm số lượng nhiều nhất, khoảng từ 4.2 triệu đến 4.5 triệu/mm³ máu đối với nữ và 4.5 triệu đến 5.0 triệu/mm³ máu đối với nam. Hồng cầu chứa huyết sắc tố (chất làm cho máu có màu đỏ) là chất vận chuyển oxy từ phổi đến các mô và nhận khí CO₂ từ các mô lên đào thải ở phổi. Hồng cầu sống tối đa là 120 ngày, trung bình là 90 ngày. Hồng cầu già sẽ bị tiêu hủy chủ yếu ở lách và gan. Tủy xương sẽ sinh các hồng cầu mới thay thế để duy trì lượng hồng cầu ổn định trong cơ thể.

1.1.1.2. Bạch cầu

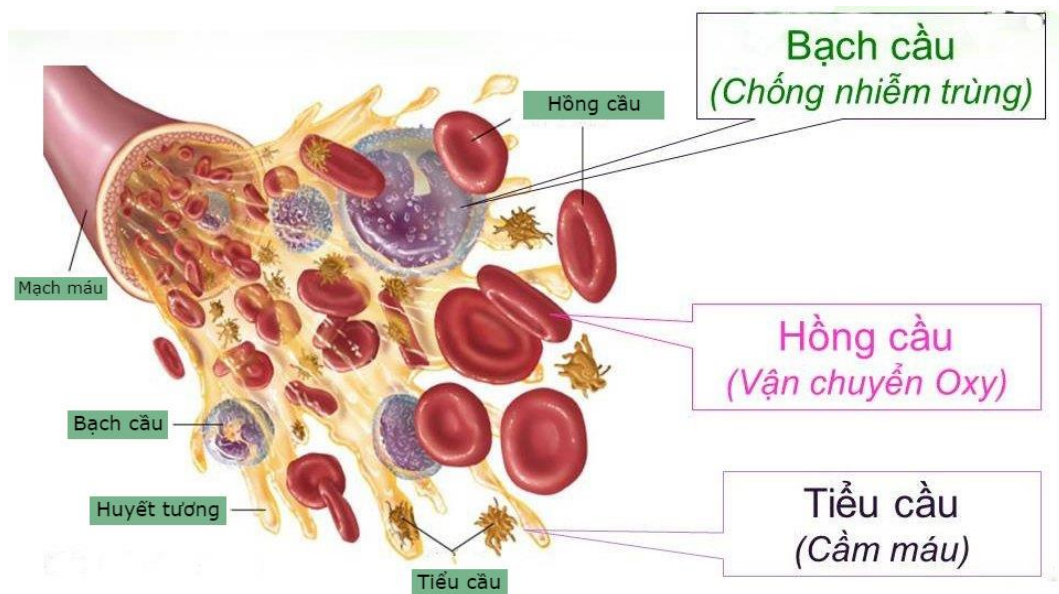
Bạch cầu là tế bào to, có nhân, chiếm số lượng từ 4000 đến 10000/mm³ máu. Bạch cầu có chức năng bảo vệ cơ thể bằng cách phát hiện và tiêu diệt các “vật lạ” gây bệnh. Có nhiều loại bạch cầu khác nhau với đời sống từ một tuần đến vài tháng, có loại làm nhiệm vụ thực bào tức là “ăn” các vật lạ gây bệnh, có loại làm nhiệm vụ “nhớ” để nếu lần sau vật lạ này xâm nhập sẽ bị phát hiện và nhanh chóng cơ thể sinh ra một lượng lớn bạch cầu tiêu diệt chúng, có loại tiết ra các kháng thể lưu hành trong máu để bảo vệ cơ thể,... Bạch cầu được sinh ra tại tủy xương. Bình thường ngoài lưu hành trong máu, một lượng lớn bạch cầu cư trú ở các mô của cơ thể để làm nhiệm vụ bảo vệ.

Bạch cầu được chia thành năm loại với các tỉ lệ như sau:

- Bạch cầu đa nhân trung tính (neutrophil): tỉ lệ $68 \pm 5 \%$
- Bạch cầu ái toan (eosinophil): $2 \pm 2 \%$
- Bạch cầu ái kiềm (basophil): 0.4%
- Bạch cầu đơn nhân (monocyte): $1.4 \pm 0.4 \%$
- Bạch cầu lympho (lymphocyte): $27 \pm 5 \%$

1.1.1.3. Tiểu cầu

Tiểu cầu là những mảnh tế bào rất nhỏ, số lượng từ 150 nghìn đến 350 nghìn/mm³ máu. Tiểu cầu tham gia vào chức năng cầm máu, tham gia tạo các cục máu đông bịt các vết thương thành mạch. Ngoài ra, tiểu cầu còn làm cho thành mạch mềm mại, dẻo dai nhờ chức năng tiểu cầu làm “trẻ hóa” thành mạch. Tiểu cầu sống được từ một đến hai tuần, và cũng được sinh ra từ tủy xương.



Hình 1.2 Các thành phần của tế bào máu

1.1.2 Thành phần huyết tương

Huyết tương là một phần của máu và chứa nhiều loại protein và chất dinh dưỡng như albumin, globulin và fibrinogen. Albumin là một protein quan trọng trong huyết tương và có chức năng duy trì áp lực máu và cân bằng nước trong cơ thể. Globulin là một loại protein chuyển hóa trong quá trình miễn dịch và có chức năng bảo vệ cơ thể chống lại bệnh tật. Fibrinogen là một chất đóng góp vào quá trình đông máu để ngăn chặn chảy máu khi có tổn thương.

Huyết tương theo các mạch máu đến ruột hấp thu các chất dinh dưỡng rồi đi nuôi khắp cơ thể. Đồng thời nhận các chất cần đào thải từ các mô, các tế bào để đưa đến chuyển hóa ở gan, đào thải ở thận, tuyến mồ hôi, phổi,... Huyết tương thay đổi theo giờ trong cơ thể, ví dụ sau bữa ăn huyết tương có màu đục và trở lên trong, màu vàng chanh sau khi ăn từ một đến hai giờ.

1.2 Các bệnh lý máu

Các bệnh lý máu có thể được chia thành nhiều loại khác nhau, tùy thuộc vào các tế bào và thành phần máu bị ảnh hưởng.

1.2.1 Bệnh lý liên quan đến rối loạn hồng cầu

Rối loạn tạo hồng cầu được gọi là bệnh lý khi hồng cầu trong máu quá tăng hay quá giảm để đưa đến hai hậu quả chính:

- Giảm sinh hồng cầu, tức bệnh thiếu máu.
- Tăng sinh hồng cầu tăng sinh về lượng hoặc tăng sinh ác tính hóa, tức bệnh đa hồng cầu.

1.2.1.1. Thiếu máu

Thiếu máu là sự giảm Hb trong máu dưới những giá trị bình thường.

Theo tổ chức Y tế thế giới, người bị thiếu máu là người có nồng độ Hb giảm:

- Ở nam giới: < 13 gam Hb/100 ml máu
- Ở nữ giới: < 12 gam Hb/100 ml máu
- Ở trẻ sơ sinh: < 14 gam Hb/100 ml máu

Thiếu máu có thể do sự mất máu quá nhanh hoặc do sự sản xuất hồng cầu quá chậm.

1.2.1.2. Đa hồng cầu

Đa hồng cầu thứ phát: Do các mô bị thiếu oxy sẽ kích thích thận và gan sản xuất ra erythropoietin thúc đẩy quá trình tạo hồng cầu ở tủy xương.

Nguyên nhân gây ra thiếu oxy ở mô có thể là do sống ở vùng cao, suy tim, các bệnh đường hô hấp,... những điều kiện này sẽ gây tăng hồng cầu thứ phát và số lượng hồng cầu có thể tăng lên tới 6-8 triệu/mm³ máu [2].

Một số nguyên nhân hay gặp nhất của tăng hồng cầu thứ phát gọi là tăng hồng cầu sinh lý của những dân tộc sống ở độ cao 4000-5000 mét, những người này có số lượng hồng cầu là 6-8 triệu/ mm³ máu.

Tăng hồng cầu thực sự: Trong bệnh này tủy xương sản xuất ra quá nhiều hồng cầu, số lượng bạch cầu và tiểu cầu cũng tăng. Những người bị bệnh tăng hồng cầu thực sự có số lượng hồng cầu từ 7-8 triệu/ mm³. Hematocrit 60 đến 70% và thể tích máu cũng tăng, có thể gấp đôi bình thường. Hậu quả là hệ thống mạch bị quá tải, một số mao mạch bị bít kín do độ quánh của máu tăng gấp ba lần bình thường [2].

1.2.2 Bệnh lý liên quan đến rối loạn tiểu cầu

Rối loạn tiểu cầu có thể gây ra các bệnh lý khác nhau, từ bệnh lý miễn dịch đến bệnh lý di truyền. các bệnh lý về rối loạn tiểu cầu được nghiên cứu là:

- Tăng tiểu cầu bất thường (tăng tiểu cầu nguyên phát và thứ phát)
- Giảm tiểu cầu
- Rối loạn chức năng tiểu cầu

Đối với tăng tiểu cầu nguyên phát: Đây là bệnh lý do tế bào tiểu cầu tăng vượt quá mức bình thường do tế bào tiểu cầu khỏe mạnh tự động phân chia và phát triển một cách không kiểm soát. Bệnh này thường không gây ra triệu chứng và được phát hiện trong các kết quả xét nghiệm máu thường xuyên. Để chẩn đoán bệnh tăng tiểu cầu nguyên phát, các bác sĩ thường sử dụng các phương pháp như xét nghiệm máu, xét nghiệm tủy xương và kiểm tra các chỉ số chức năng gan.

Đối với tăng tiểu cầu thứ phát: Đây là bệnh lý do tế bào tiểu cầu tăng vượt quá mức bình thường do phản ứng của hệ miễn dịch với các bệnh lý khác như viêm nhiễm hoặc ung thư. Bệnh này thường gây ra các triệu chứng như sốt, đau và khó thở. Để chẩn đoán bệnh tăng tiểu cầu thứ phát, các bác sĩ sẽ thực hiện các xét nghiệm để xác định nguyên nhân gây bệnh.

Đối với giảm tiểu cầu: Giảm tiểu cầu là tình trạng tiểu cầu trong máu dưới mức bình thường. Giảm tiểu cầu có thể do nhiều nguyên nhân khác nhau, bao gồm bệnh lý của tủy xương, tác dụng của một số thuốc hoặc các bệnh lý miễn dịch khác. Các triệu chứng của giảm tiểu cầu bao gồm suy nhược, chảy máu dưới da và dễ bị nhiễm trùng. Để chẩn đoán giảm tiểu cầu, các

bác sĩ thường sử dụng các phương pháp như xét nghiệm máu, xét nghiệm tủy xương và kiểm tra chức năng gan và thận. Nếu giảm tiểu cầu do bệnh lý tủy xương, bệnh nhân có thể được điều trị bằng các phương pháp như truyền tế bào gốc hoặc xương sống. Nếu giảm tiểu cầu do tác dụng phụ của thuốc, bệnh nhân có thể được chuyển sang thuốc khác để điều trị.

Rối loạn chức năng tiểu cầu: có thể xuất phát từ sự khiếm khuyết tại tiểu cầu hoặc từ một yếu tố bên ngoài thay đổi chức năng bình thường của tiểu cầu. Rối loạn chức năng tiểu cầu có thể làm cho tiểu cầu không thực hiện được chức năng của nó, gây ra các vấn đề sức khỏe. Các rối loạn chức năng tiểu cầu bao gồm các bệnh lý như thalassemia và bệnh von Willebrand. Để chẩn đoán các rối loạn chức năng tiểu cầu, các bác sĩ sử dụng các xét nghiệm đặc biệt để xác định các chỉ số chức năng của tiểu cầu.

1.2.3 Bệnh lý liên quan đến rối loạn bạch cầu

Tạo bạch cầu được coi là bất thường về số lượng khi số lượng bạch cầu trong máu lệch khỏi các giá trị sinh lý. Có hai tình trạng chính là tăng hoặc giảm số lượng bạch cầu. Để xác định loại bạch cầu nào gây ra sự biến đổi này, cần phải dựa vào tỉ lệ phần trăm của mỗi loại bạch cầu. Vì vậy, xét nghiệm đếm bạch cầu luôn đi kèm với xét nghiệm công thức bạch cầu. Ngoài ra, trong nhiều trường hợp, cần phải phân tích cả sự thay đổi về chất của bạch cầu thông qua mô tả hình thái và các xét nghiệm miễn dịch, hóa mô.

Bệnh rối loạn tạo bạch cầu thường được phân loại như sau:

- Tăng bạch cầu: bao gồm có hồi phục và không hồi phục.
- Giảm bạch cầu: bao gồm có hồi phục và không hồi phục.
- Thay đổi về chất: Suy giảm chức năng bạch cầu; bệnh leukemia và các dị sản khác.

1.2.3.1. Tăng bạch cầu

Xét nghiệm đếm bạch cầu ở máu cho con số bình thường là $6000-8000/\text{mm}^3$, một số kết quả điều tra còn có thể mở rộng giới hạn bình thường tới $5000-9000/\text{mm}^3$. Dựa vào tỉ lệ phần trăm của mỗi dòng bạch cầu, sẽ tính được số lượng tuyệt đối từng dòng. Đây mới là con số thật sự cho biết một dòng ở máu ngoại vi là tăng hay giảm số lượng. Tuy nhiên, chưa hẳn con số này đã phản ánh trung thực sự sinh sản bạch cầu toàn cơ thể.

Một dòng bạch cầu có số lượng tuyệt đối bị giảm ở máu có thể do kém sản xuất, nhưng cũng có thể do tăng hủy hoại (dù sản xuất vẫn tăng) [2].

- Tăng bạch cầu có hồi phục

Đó là hiện tượng tăng nhưng vẫn tự có khả năng trở về bình thường. Không kể các tình trạng tăng bạch cầu trong ăn uống, xúc cảm, hoạt động thể lực thì tăng bạch cầu hay gặp nhất là trong viêm, phổ biến là viêm do nhiễm khuẩn và ngay sau đó là viêm do dị ứng. Cuối cùng, có thể gặp tăng bạch cầu trong những bệnh phát triển ác tính của chính mô bạch cầu [2].

- Tăng bạch cầu không hồi phục (hay còn gọi là bệnh leukemia)

Đó là bệnh tăng số lượng bạch cầu, với đặc điểm là tăng rất cao và chỉ có khả năng giảm tạm thời, ngắn hạn, cho đến khi tử vong. Bệnh hay gặp nhất có tên leukemia, trong đó bệnh nhân bị bệnh có bạch cầu tăng rất cao về lượng và thay đổi rất nhiều về chất, bệnh còn có tên leucose, “bệnh hạch huyết”, hay bệnh ung thư bạch cầu. Thực chất, đây là sự phát triển ác tính của một dòng bạch cầu. Đến nay, khái niệm mở rộng sang cả dòng tiểu cầu và hồng cầu [2].

Phân loại bệnh theo 2 tiêu chuẩn:

- Nơi bị bệnh: là tủy hay mô lympho, từ đó phân thành hai nhóm leukemia nguồn gốc tủy, leukemia nguồn gốc lympho.

- Cấp tính hay mãn tính: Dựa vào hình thái tế bào non (chưa biệt hóa hay ít biệt hóa) hay già, không hẳn là diễn biến nhanh hay kéo dài. Tuy nhiên, trên thực tế, nếu sự ác hóa rơi vào giai đoạn non thì bệnh thường diễn biến nhanh, cấp, tiên lượng xấu; ngược lại, cũng hay gặp bệnh diễn biến chậm, kéo dài, nếu sự ác tính hóa xảy ra khi tế bào đã biệt hóa cao.

Hầu hết trường hợp bệnh leukemia có tăng bạch cầu, do vậy nó được xếp vào nhóm tăng bạch cầu. Tuy nhiên, đôi khi cá biệt có số lượng bạch cầu bình thường, hoặc giảm bạch cầu. Dù vậy, lâm sàng và chất lượng bạch cầu đều cho thấy đây là bệnh ác tính [2].

1.2.3.2. Giảm bạch cầu

Bệnh giảm bạch cầu được gặp tại hai trường hợp là giảm tổng số bạch cầu và giảm bạch cầu theo từng dòng. Giảm tổng số bạch cầu là khi bạch cầu ở máu chỉ còn dưới 5000 cho mỗi mm^3 , có nghiên cứu đưa ra mức dưới 3000.

Trường hợp nặng nhất gọi là tuyệt sản bạch cầu, nếu số lượng dưới 1000 và hoàn toàn không có thành phần trẻ ở máu ngoại vi. Giảm từng dòng bạch cầu khi các dòng khác vẫn có số lượng bình thường, hoặc còn tăng lên.

Muốn có kết luận về giảm dòng nào phải tính ra số lượng tuyệt đối của dòng đó mà không chỉ dựa vào tỉ lệ phần trăm của nó. Ví dụ, trong nhiễm HIV, tổng số lympho bào khá lâu vẫn nằm trong giới hạn bình thường nhưng có thể lượng tế bào T, và nhất là CD4 đã bắt đầu giảm thật sự [2].

1.2.3.3. *Thay đổi về chất của bạch cầu*

Suy giảm chức năng bạch cầu: Đây là tình trạng các tế bào bạch cầu không thực hiện được chức năng của nó, gây ra các vấn đề sức khỏe. Suy giảm chức năng bạch cầu thường gây ra các triệu chứng như nhiễm trùng nặng và dễ tái phát.

Bệnh leukemia và các dị sản khác: Đây là tình trạng một số tế bào bạch cầu trở nên bất thường và phát triển không kiểm soát. Bệnh này gây ra các triệu chứng như suy nhược, chảy máu và nhiễm trùng dễ xảy ra. Để chẩn đoán bệnh leukemia và các dị sản khác, các bác sĩ sử dụng các phương pháp như xét nghiệm máu, xét nghiệm tủy xương và các phương pháp hình ảnh để xác định sự bất thường của tế bào bạch cầu.

1.3 Mục đích và ý nghĩa của việc phân loại tế bào máu

Việc phân loại tế bào máu đóng vai trò quan trọng trong việc chẩn đoán, theo dõi và điều trị các bệnh lý máu:

Chẩn đoán bệnh lý máu: Bằng cách phân tích và phân loại các tế bào máu, bác sĩ có thể xác định các bệnh lý máu mà bệnh nhân có thể mắc phải, chẳng hạn như thiếu máu, bệnh lý rối loạn tiểu cầu, bệnh lý rối loạn hồng cầu, hoặc bệnh lý liên quan đến tế bào bạch cầu. Điều này giúp bác sĩ đưa ra chẩn đoán chính xác và lựa chọn phương pháp điều trị phù hợp.

Đánh giá hiệu quả điều trị: Việc phân loại tế bào máu cũng giúp bác sĩ theo dõi sự thay đổi của các chỉ số máu trong quá trình điều trị, từ đó đánh giá hiệu quả điều trị và điều chỉnh phương pháp điều trị nếu cần thiết.

Phát hiện sớm các biến chứng: Một số biến chứng của các bệnh lý máu có thể được phát hiện thông qua việc phân loại tế bào máu, giúp bác sĩ phát hiện sớm và ngăn ngừa các biến chứng tiềm ẩn.

Việc phân loại tế bào máu cũng có ý nghĩa trong nghiên cứu khoa học, giúp các nhà nghiên cứu tìm hiểu về cấu trúc, chức năng, và tương tác của các loại tế bào máu, cũng như phát triển các phương pháp chẩn đoán và điều trị mới cho các bệnh lý máu.

Phân loại tế bào máu cũng là một phần của các xét nghiệm máu tổng quát, giúp đánh giá sức khỏe tổng thể của một người và phát hiện sớm các dấu hiệu bất thường trong hệ thống tuần hoàn máu. Điều này giúp người bệnh và bác sĩ đưa ra các biện pháp phòng ngừa và điều trị kịp thời khi phát hiện các vấn đề sức khỏe tiềm ẩn.

Tăng cường chất lượng chăm sóc sức khỏe: Việc phân loại tế bào máu và đánh giá chính xác các bệnh lý máu giúp tăng cường chất lượng chăm sóc sức khỏe cho người bệnh, hạn chế các sai sót trong quá trình chẩn đoán và điều trị, đồng thời giảm thiểu tác động tiêu cực đến chất lượng cuộc sống của người bệnh.

Hỗ trợ quyết định lâm sàng: Việc phân loại tế bào máu cung cấp thông tin quan trọng để hỗ trợ quyết định lâm sàng của bác sĩ, giúp họ lựa chọn các biện pháp chẩn đoán và điều trị phù hợp với từng trường hợp cụ thể.

1.4 Các phương pháp phân loại tế bào máu đang được sử dụng trong bệnh viện

1.4.1 Phương pháp tế bào học dòng chảy

Phương pháp tế bào học dòng chảy (Flow Cytometry) là một kỹ thuật quan trọng trong việc chẩn đoán các bệnh lý máu. Kỹ thuật này cho phép đánh giá đồng thời nhiều thông tin về từng tế bào trong mẫu máu thông qua việc phân tích tính chất quang học của tế bào khi chúng chảy qua một dòng chảy mỏng. Thông qua việc sử dụng các đánh dấu (marker) đặc trưng, Phương pháp tế bào học dòng chảy có khả năng xác định các thành phần tế bào máu, giúp phân loại và định lượng các loại tế bào khác nhau trong một mẫu máu.

Trong phương pháp này, mẫu máu được chuẩn bị bằng cách gắn các đánh dấu đặc trưng với các loại tế bào máu khác nhau. Sau đó, mẫu được đưa vào một thiết bị dòng chảy, trong đó tế bào máu sẽ chảy qua một dòng chảy mỏng và được phơi sáng với ánh sáng laser. Các đánh dấu sẽ phát quang khi tiếp xúc với ánh sáng, tạo ra những tín hiệu quang học mà máy đọc và phân tích. Từ đó, phân tích dữ liệu thu

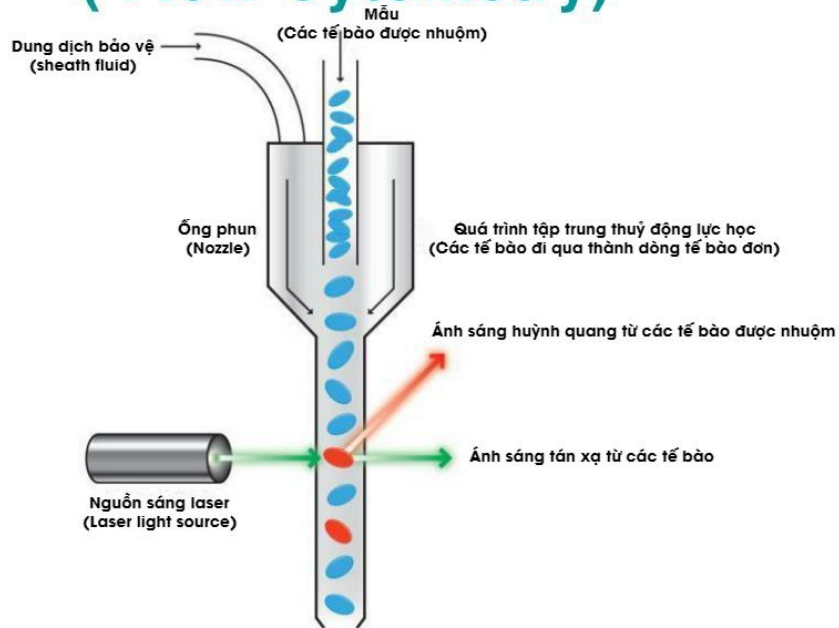
được sẽ cho ta thông tin về số lượng, kích thước, độ phức tạp và đặc điểm biểu hiện của các đánh dấu trên bề mặt tế bào.

Phương pháp tế bào học dòng chảy có nhiều ưu điểm so với các phương pháp khác. Đầu tiên, nó cho phép định lượng và phân loại các loại tế bào máu nhanh chóng, chính xác và đồng thời. Thứ hai, kỹ thuật này có độ nhạy cao, giúp phát hiện được những tế bào bất thường dù chỉ chiếm tỷ lệ rất nhỏ trong mẫu máu. Thứ ba, phương pháp này không chỉ giúp phân loại các loại tế bào máu mà còn có thể cung cấp thông tin về hoạt động của các tế bào, chẳng hạn như phân hủy tế bào hay hoạt động miễn dịch.

Tuy nhiên, phương pháp tế bào học dòng chảy cũng có một số hạn chế. Đầu tiên, chi phí thiết bị và vật tư tiêu hao cho kỹ thuật này thường khá cao, đặc biệt là khi sử dụng nhiều đánh dấu khác nhau. Điều này có thể làm tăng chi phí chẩn đoán cho các bệnh nhân và các cơ sở y tế. Thứ hai, kỹ thuật này đòi hỏi độ chính xác cao trong quá trình chuẩn bị mẫu và thực hiện phân tích, do đó yêu cầu kỹ năng và kinh nghiệm của người thực hiện. Thứ ba, dù phương pháp này có độ nhạy cao, việc phân tích dữ liệu thu được có thể bị ảnh hưởng bởi nhiễu tín hiệu hoặc sai số trong quá trình đọc dữ liệu.

Phương pháp tế bào học dòng chảy có hàng loạt các ứng dụng quan trọng trong nghiên cứu và chẩn đoán lâm sàng. Những nhóm ứng dụng lớn nhất có thể kể đến như xác định sự biểu hiện của các thụ thể trên bề mặt tế bào trong đếm số lượng tế bào miễn dịch TCD-4, TCD-8, Tế bào giết tự nhiên NK... trong chẩn đoán theo dõi các bệnh suy giảm miễn dịch PIDs, HIV, chẩn đoán các bệnh máu ác tính, đánh giá tồn dư bệnh tối thiểu (MRD - Minimal Residual Disease), các bệnh lý tiểu cầu, phân tích các quần thể tế bào đơn như tốc độ tăng trưởng (proliferation), sự chết theo chương trình (apoptosis), chức năng, tín hiệu tế bào [8]

Nguyên tắc đếm tế bào dòng chảy (Flow Cytometry)



Hình 1.3 Nguyên tắc đếm tế bào dòng chảy

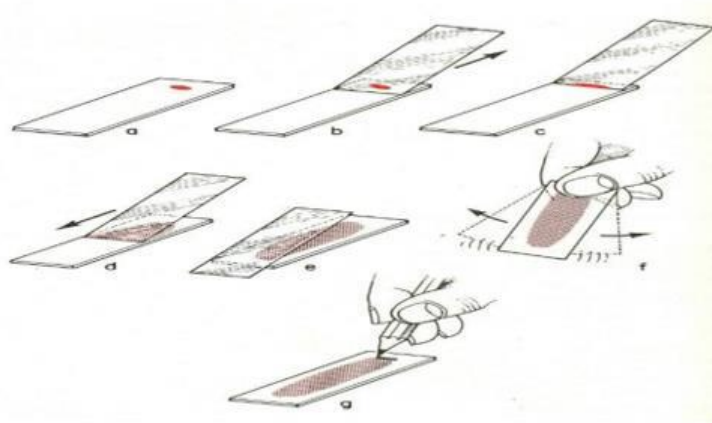
1.4.2 Phương pháp soi mẫu phẩm dưới kính hiển vi

Phương pháp soi mẫu phẩm dưới kính hiển vi là một trong những kỹ thuật cổ điển và phổ biến nhất trong việc chẩn đoán các bệnh lý máu. Kỹ thuật này dựa trên việc sử dụng kính hiển vi để quan sát các tế bào máu trong mẫu phẩm, từ đó phân loại và đánh giá các đặc điểm của chúng. Dưới đây là mô tả chi tiết về quy trình chuẩn bị mẫu, phương pháp soi và ứng dụng của kỹ thuật này trong chẩn đoán các bệnh lý máu.

- *Chuẩn bị mẫu phẩm:* Để thực hiện phương pháp này, mẫu máu cần được lấy từ bệnh nhân và được chế tác thành một mẫu phẩm mỏng trên một đế kính. Mẫu phẩm sau đó được nhuộm bằng các loại thuốc nhuộm đặc biệt, giúp tăng độ tương phản giữa các thành phần của tế bào và nền. Một số thuốc nhuộm phổ biến trong phân loại tế bào máu bao gồm thuốc nhuộm Wright, Giemsa và May-Grunwald. Các thuốc nhuộm này giúp đánh dấu các thành phần trong tế bào máu, như hạt nhân, tiểu nhân và hạt chất lượng, giúp phân biệt giữa các loại tế bào khác nhau.
- *Soi mẫu phẩm dưới kính hiển vi:* Sau khi chuẩn bị xong mẫu phẩm, các tế bào máu sẽ được quan sát dưới kính hiển vi ở độ phóng đại cao (thường là

1000x). Người thực hiện sẽ đánh giá các đặc điểm của tế bào, như kích thước, hình dạng, tỷ lệ và vị trí của hạt nhân, cũng như sự hiện diện của các hạt chất lượng và tiểu nhân. Dựa trên các đặc điểm này, các tế bào máu sẽ được phân loại thành các loại tế bào khác nhau, như tế bào hồng cầu, tế bào bạch cầu và tiểu cầu.

- Đánh giá kết quả: Kết quả của phương pháp soi mẫu phẩm dưới kính hiển vi sẽ được đánh giá dựa trên các tiêu chí như số lượng và tỷ lệ các loại tế bào máu, cũng như các đặc điểm bất thường của chúng. Bằng cách so sánh kết quả này với các giá trị chuẩn của người bình thường, các bác sĩ có thể xác định các bệnh lý máu mà bệnh nhân có thể mắc phải, chẳng hạn như thiếu máu, bệnh lý rối loạn tiểu cầu, bệnh lý rối loạn hồng cầu, hoặc bệnh lý liên quan đến tế bào bạch cầu.



Hình 1.4 Quy trình thực hiện phết máu ngoại vi

Ưu điểm và nhược điểm của phương pháp soi mẫu phẩm dưới kính hiển vi:

- Ưu điểm:
 - Cho phép quan sát trực tiếp các tế bào máu, giúp xác định chính xác các đặc điểm của chúng.
 - Có chi phí thấp và dễ tiếp cận, khiến nó trở thành một công cụ chẩn đoán phổ biến trong các cơ sở y tế.
 - Cung cấp kết quả nhanh chóng, giúp bác sĩ đưa ra các quyết định điều trị kịp thời.
- Nhược điểm:
 - Đòi hỏi kỹ năng và kinh nghiệm của người thực hiện, do đó dẫn đến sự không nhất quán giữa các kết quả của các bác sĩ khác nhau.

- Có độ nhạy thấp hơn so với các phương pháp khác, chẳng hạn như phương pháp tế bào học dòng chảy, khiến nó khó phát hiện được các tế bào bất thường khi tỷ lệ của chúng trong mẫu máu thấp.
- Việc phân loại tế bào máu dưới kính hiển vi là một quá trình thủ công, tốn nhiều thời gian và công sức của bác sĩ, đặc biệt khi phải đối mặt với số lượng lớn mẫu phẩm.

Từ những nhược điểm trên việc tìm kiếm và phát triển các phương pháp phân loại tế bào máu tự động, ít tốn kém hơn và dễ dàng tiếp cận hơn đang trở thành một nhu cầu thiết yếu. Một số phương pháp phân loại tế bào máu tự động sử dụng thuật toán và phần mềm đã được nghiên cứu và phát triển, nhằm giảm thiểu sự phụ thuộc vào kỹ năng và kinh nghiệm của người thực hiện, đồng thời tăng độ chính xác và độ nhạy của quá trình phân loại tế bào máu.

1.4.3 Một số phương pháp phân loại tế bào máu tự động

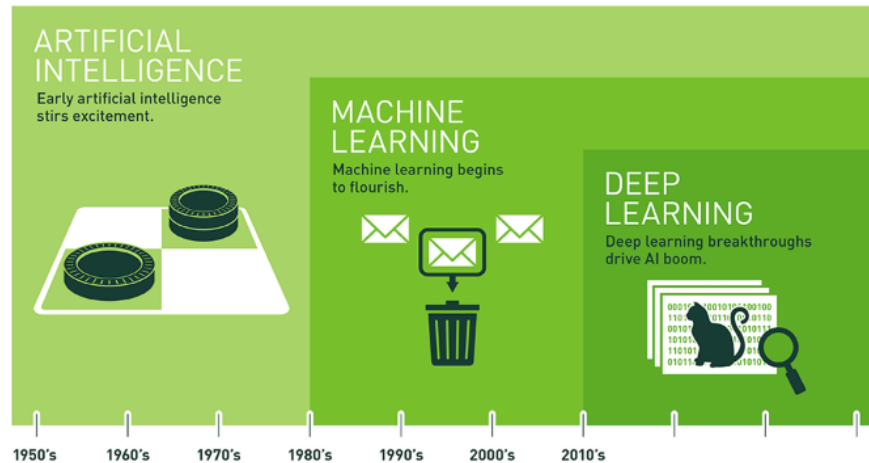
Với sự bùng nổ của khoa học máy tính đặc biệt là lĩnh vực trí tuệ nhân tạo, con người đã xây dựng và phát triển kỹ thuật học máy machine learning:

Machine learning (ML) (học máy) là việc phát triển các thuật toán và mô hình để máy tính có thể tự học và cải thiện kỹ năng thực hiện một nhiệm vụ cụ thể mà không cần được lập trình cụ thể cho nhiệm vụ đó. Mục tiêu của học máy là tạo ra các mô hình dựa trên dữ liệu huấn luyện, giúp máy tính dự đoán, phân loại hoặc nhận biết mẫu từ dữ liệu mới mà chưa từng gặp. Học máy được sử dụng trong nhiều lĩnh vực khác nhau, chẳng hạn như nhận dạng giọng nói, kiểm soát chất lượng, phát hiện gian lận, dự báo giá cả, phân loại văn bản và hình ảnh, xe tự lái, đề xuất sản phẩm.

Học máy có ba loại chính:

- Học có giám sát (Supervised learning): Trong học có giám sát, thuật toán được huấn luyện trên một tập dữ liệu gồm các cặp đầu vào và đầu ra mong muốn. Mục tiêu là tìm ra một mối quan hệ giữa đầu vào và đầu ra để có thể dự đoán kết quả cho dữ liệu mới.
- Học không giám sát (Unsupervised learning): Trong học không giám sát, thuật toán được huấn luyện trên tập dữ liệu chỉ gồm đầu vào mà không có đầu ra mong muốn. Mục tiêu là tìm ra cấu trúc ẩn, nhóm hoặc mối quan hệ giữa các điểm dữ liệu.

- Học tăng cường (Reinforcement learning): Trong học tăng cường, một "tác tử" học cách thực hiện hành động trong một môi trường sao cho đạt được điểm thưởng tối đa. Tác tử không nhận được hướng dẫn trực tiếp về hành động nào là tốt nhất, mà phải khám phá và học từ kinh nghiệm.



Hình 1.5 Mối quan hệ giữa AI, ML, DL

Từ nền tảng của ML kết hợp với sự phát triển của mạng neural đã phát triển công nghệ học sâu deep learning.

Deep learning (DL) là một nhánh của học máy, tập trung vào việc xây dựng và huấn luyện các mạng neuron nhân tạo (Artificial Neural Networks - ANN) có nhiều lớp ẩn, được gọi là mạng neural sâu (Deep Neural Networks - DNN). Deep learning giúp máy tính tự học và trích xuất đặc trưng từ dữ liệu phức tạp và lớn mà không cần phải xác định rõ ràng các đặc trưng trước đó.

Mạng neural sâu bao gồm nhiều lớp neural, trong đó mỗi lớp chịu trách nhiệm học một phần của biểu diễn dữ liệu. Lớp đầu tiên thường học các đặc trưng cơ bản nhất (ví dụ: đường nét trong ảnh), trong khi các lớp sau học các đặc trưng ngày càng phức tạp và trừu tượng hơn. Sự kết hợp của nhiều lớp giúp mạng neural sâu có khả năng biểu diễn và học được các mẫu phức tạp từ dữ liệu.

Deep learning đã tạo ra những bước đột phá đáng kể trong nhiều lĩnh vực, như nhận dạng hình ảnh, giọng nói, dịch máy và hiểu ngôn ngữ tự nhiên. Một số ứng dụng nổi bật của deep learning bao gồm:

- Mạng neuron chập (Convolutional Neural Networks - CNN) - được sử dụng rộng rãi trong phân loại hình ảnh và nhận dạng vật thể.

- Mạng neuron hồi quy (Recurrent Neural Networks - RNN) và các biến thể của nó như Long Short-Term Memory (LSTM) - được sử dụng trong dự báo chuỗi thời gian, tổng hợp văn bản và hiểu ngôn ngữ tự nhiên.
- Mạng neuron hồi quy (Recurrent Neural Networks - RNN) và các biến thể của nó như Long Short-Term Memory (LSTM) - được sử dụng trong dự báo chuỗi thời gian, tổng hợp văn bản và hiểu ngôn ngữ tự nhiên.

Từ công nghệ học sâu DL, con người đã ứng dụng và phát triển bài toán nhận diện vật thể trong ảnh và đưa ra vị trí của vật (object detection) với hai phương pháp phát hiện là Two-stage object detectors và One-stage object detectors.

1.4.3.1. *Two-stage object detectors*

Two-stage object detectors (bộ phát hiện đối tượng hai giai đoạn) là một loại phương pháp phát hiện và xác định vị trí của các đối tượng trong hình ảnh hoặc video, dựa trên cách tiếp cận hai giai đoạn. Hai giai đoạn trong bộ phát hiện đối tượng bao gồm việc tạo ra các khung chứa (region proposals) và phân loại chúng.

- Giai đoạn 1: Tạo khung chứa - Bộ phát hiện đối tượng sẽ tìm kiếm các vùng tiềm năng trong hình ảnh có thể chứa đối tượng. Để làm điều này, mô hình sẽ sử dụng một thuật toán như Selective Search, EdgeBoxes hoặc Region Proposal Network (RPN) để tạo ra một tập hợp các ứng viên khung chứa (bounding box proposals) dựa trên các đặc trưng hình ảnh và cấu trúc.
- Giai đoạn 2: Phân loại và hiệu chỉnh khung chứa - Các khung chứa từ giai đoạn 1 sẽ được đưa vào một mạng neuron chập (Convolutional Neural Network - CNN) hoặc một mạng tương tự để phân loại và xác định vị trí chính xác hơn. Mạng này sẽ xác định lớp của đối tượng trong mỗi khung chứa và hiệu chỉnh kích thước và vị trí của khung chứa để tối ưu hóa sự khớp giữa khung chứa và đối tượng thực tế.

Một ví dụ nổi tiếng về two-stage object detectors là R-CNN (Regions with CNN features) và các biến thể của nó, như Fast R-CNN và Faster R-CNN. Các phương pháp này đã đạt được kết quả tốt trong việc phát hiện đối tượng và được sử dụng rộng rãi trong nhiều ứng dụng.

Ứng dụng vào bài toán phân loại tế bào sử dụng mô hình two-stage object detectors có một số bài báo sau:

Bảng 1.1 Nghiên cứu phân loại tế bào bạch cầu theo mô hình two stage detector

Tác giả	Năm	Mô hình nghiên cứu
Banik et al	2020	CNN with fusing features in the first and last convolutional layer
Basnet et al	2020	DCNN model with image pre-processing and a modified loss function
Baydilli et al.	2020	WBC classification using a small dataset via capsule networks
Cinar et al	2021	Hybrid AlexNet, GoogleNet networks, and support vector machine
Khan et al	2021	AlexNet model with feature selection strategy and extreme learning machine (ELM)
Xiwei Huang	2021	High-Efficiency Classification of White Blood Cells Based on Object Detection
Wang et al	2021	Three-dimensional attention networks for hyperspectral images

1.4.3.2. One-stage object detectors

One-stage object detectors (bộ phát hiện đối tượng một giai đoạn) là một loại phương pháp phát hiện và xác định vị trí của các đối tượng trong hình ảnh hoặc video mà chỉ cần một giai đoạn xử lý. Trong one-stage object detectors, việc phân loại và xác định vị trí đối tượng được thực hiện đồng thời trong một lần xử lý, giúp cải thiện tốc độ so với các phương pháp hai giai đoạn (two-stage object detectors).

Có hai mô hình one-stage object detectors nổi tiếng:

- YOLO (You Only Look Once): YOLO chia hình ảnh thành lưới ô vuông và sử dụng một mạng CNN để dự đoán vị trí, kích thước, và loại đối tượng trong mỗi ô của lưới. YOLO được biết đến với tốc độ xử lý nhanh và khả năng phát hiện đối tượng tốt trong thời gian thực.
- SSD (Single Shot MultiBox Detector): SSD cũng sử dụng một mạng CNN để dự đoán vị trí và loại đối tượng, nhưng áp dụng nhiều lớp tích

chập với kích thước khác nhau để phát hiện đối tượng ở nhiều tỷ lệ kích thước. SSD kết hợp các đặc điểm của YOLO và mạng R-CNN, mang lại hiệu suất phát hiện tốt hơn YOLO đối với các đối tượng có kích thước nhỏ.

Ứng dụng one-stage object detectors vào phân loại tế bào máu có một số bài báo như sau:

Bảng 1.2 Các nghiên cứu phân loại tế bào bạch cầu theo mô hình one stage detector

Tác giả	Năm	Mô hình nghiên cứu
Rohaziat et al	2020	White Blood Cells Detection using YOLOv3 with CNN Feature Extraction Models
Praveen et al	2021	White blood cell subtype detection and classification
Sangeetha et al	2021	An improved blood cell type detection and classsification by YOLOv4 deep learning model
Wang et al	2019	Deep learning approach to peripheral leukocyte recognition
Ramya et al	2021	Comparison of YOLOv3, YOLOv4 and YOLOv5 Performance for Detection of Blood Cells
Dương Trọng Lượng et al	2022	Yolov4 in White Blood Cell Classification

1.5 Kết luận chương

Trong chương này đã trình bày về các kiến thức cơ bản liên quan đến máu và các thành phần trong máu, nhấn mạnh vào thành phần tế bào, bao gồm hồng cầu, tiểu cầu và bạch cầu. Chúng ta đã tìm hiểu về các bệnh lý máu phổ biến và vai trò của việc phân loại tế bào máu trong việc chẩn đoán và điều trị các bệnh lý này.

Ngoài ra, chương 1 cũng đã giới thiệu các phương pháp phân loại tế bào máu hiện đại đang được sử dụng trong bệnh viện, bao gồm phương pháp tế bào học dòng chảy, phương pháp soi mẫu phẩm dưới kính hiển vi và một số phương pháp phân loại tế bào máu tự động. Chương tiếp theo, chúng ta sẽ đi sâu vào quá trình xây dựng và huấn luyện các mô hình để phát hiện và phân loại tế bào máu.

CHƯƠNG 2. XÂY DỰNG MÔ HÌNH HUẤN LUYỆN PHÁT HIỆN, PHÂN LOẠI TẾ BÀO MÁU

2.1 Phân loại tế bào bạch cầu sử dụng mô hình mạng YOLOv5

2.1.1 Sơ lược về mô hình YOLO

YOLO (You Only Look Once) là một phương pháp phát hiện đối tượng (object detection) trong hình ảnh hoặc video, nổi tiếng vì tốc độ xử lý nhanh và khả năng phát hiện đối tượng trong thời gian thực. YOLO thuộc loại one-stage object detectors, nghĩa là nó chỉ cần một lần xử lý để dự đoán vị trí và loại đối tượng trong hình ảnh.

Lịch sử dòng họ YOLO gồm:

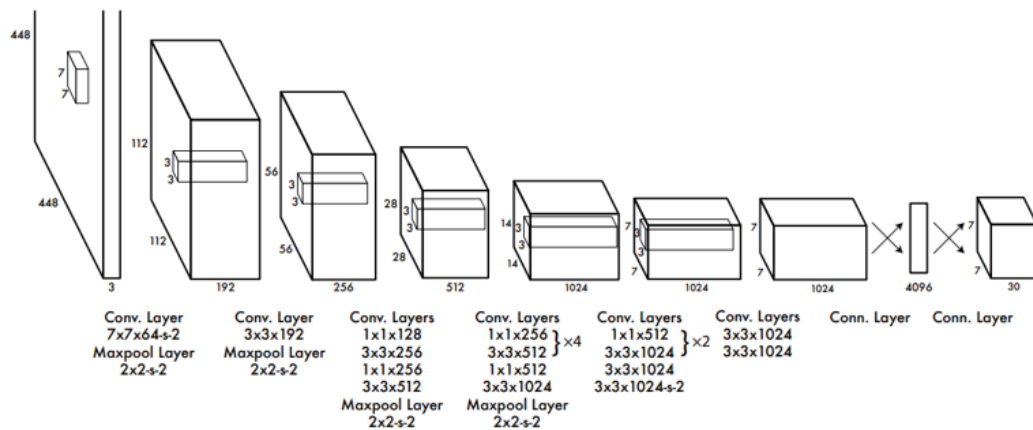
- YOLOv1 (08/06/2015): You Only Look Once: Unified, Real-Time Object Detection
- YOLOv2 (25/12/2016): YOLO9000: Better, Faster, Stronger
- YOLOv3 (08/04/2018): YOLOv3: An Incremental Improvement
- YOLOv4 (23/04/2020): YOLOv4: Optimal Speed and Accuracy of Object Detection
- YOLOv5 (18/05/2020)

Một trong những ưu điểm lớn nhất của YOLO là tốc độ xử lý nhanh, giúp nó phù hợp với các ứng dụng yêu cầu phát hiện đối tượng trong thời gian thực, như giám sát an ninh, xe tự lái, hoặc phân tích hành vi. Tuy nhiên, YOLO cũng có một số hạn chế, chẳng hạn như độ chính xác thấp hơn so với một số phương pháp hai giai đoạn (two-stage object detectors) trong một số trường hợp, đặc biệt là khi phải xử lý các đối tượng có kích thước và hình dạng khác nhau.

2.1.1.1. YOLOv1

YOLO sẽ đưa ảnh đầu vào qua một CNN, tạo ra một feature map $S \times S$, gọi là grid. YOLO thực hiện detect object tại mỗi cell (ô) trong $S \times S$ grid đó. Tức là, thay vì có một bước tìm ra các vùng có khả năng tồn tại object, thì YOLO sẽ thực hiện detect trên toàn bộ ảnh, khỏi tìm ra vùng gì cả. Một cell sẽ predict ra B Bounding Box và xác suất cho C class. Một Bounding Box sẽ mang 5 thông tin: tâm của Bounding Box (x, y), chiều dài và rộng của Bounding Box (w, h) và độ tự tin. Độ tự tin dùng để xác định xem trong Bounding Box đó có tồn tại object hay không. Nếu trong

Bounding Box đó không tồn tại object, ta bỏ qua luôn toàn bộ giá trị prediction khác của Bounding Box và Classification, nhằm bỏ qua những nơi không tồn tại object. Việc sử dụng thông tin độ tự tin này cũng phần nào giống pha thứ nhất trong phương pháp hai pha, lọc ra những nơi tồn tại object. Vậy, trong một cell, ta sẽ predict ra một tensor có $B \times 5 + C$ phần tử, với B là số lượng Bounding Box, 5 là 5 thông tin trong Bounding Box gồm: trung tâm box, chiều dài rộng box và độ tự tin và C là số class. Và trong một feature map gồm $S \times S$ cell, tensor dự đoán từ mạng sẽ có độ dài $S \times S \times (B \times 5 + C)$.



Hình 2.1 Kiến trúc mạng của YOLOv1

Hạn chế của YOLOv1:

- YOLOv1 chỉ có thể dự đoán được một object trong một cell, nếu có 2 object tồn tại trong cùng một cell, hoặc nếu phải dự đoán từng object trong một nhóm object, YOLOv1 sẽ gặp rất nhiều khó khăn.
- Với việc chỉ sử dụng 2 Bounding Box, YOLOv1 sẽ gặp khó khăn với những object có tỉ lệ khác so với trong training.
- Với việc chỉ sử dụng 2 Bounding Box, YOLOv1 sẽ gặp khó khăn với những object có tỉ lệ khác so với trong training.

2.1.1.2. YOLOv2

YOLOv1 gặp các vấn đề về Bounding Box không tốt, Recall cũng khá thấp (phát hiện được ít vật thể). Do đó YOLOv2 ra đời để cải thiện 2 vấn đề trên mà vẫn giữ được độ chính xác tốt. Bằng cách:

- Sử dụng một kiến trúc mạng mới và cách training mới.
- Áp dụng và thay đổi Anchor Box cho phù hợp.

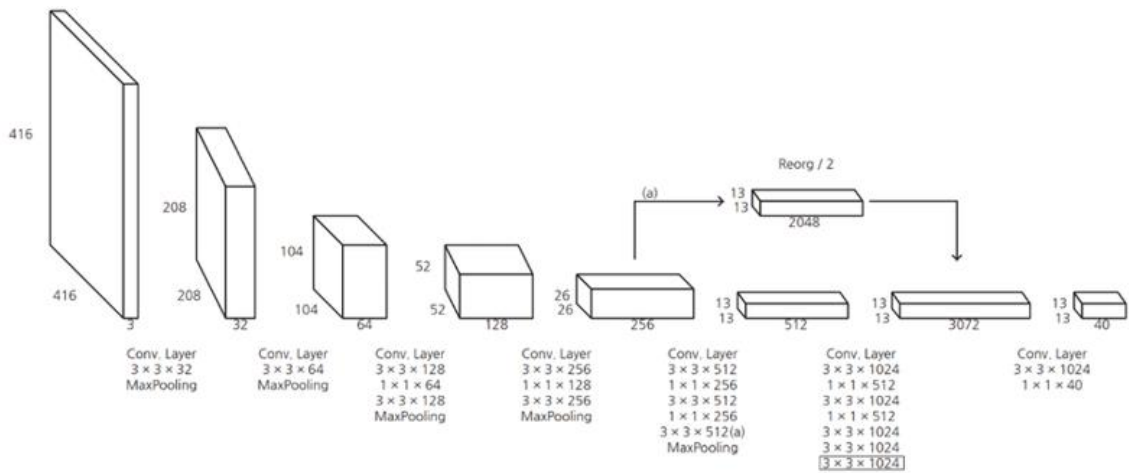
Type	Filters	Size/Stride	Output
Convolutional	32	3×3	224×224
Maxpool		$2 \times 2/2$	112×112
Convolutional	64	3×3	112×112
Maxpool		$2 \times 2/2$	56×56
Convolutional	128	3×3	56×56
Convolutional	64	1×1	56×56
Convolutional	128	3×3	56×56
Maxpool		$2 \times 2/2$	28×28
Convolutional	256	3×3	28×28
Convolutional	128	1×1	28×28
Convolutional	256	3×3	28×28
Maxpool		$2 \times 2/2$	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Maxpool		$2 \times 2/2$	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	1000	1×1	7×7
Avgpool		Global	1000
Softmax			

Hình 2.2: Kiến trúc mạng của YOLOv2

Darknet-19: Kiến trúc mạng của YOLOv2 được thể hiện ở hình 2.2, được gọi là Darknet-19 vì có 19 lớp Convolution (Conv).

Batch Norm: Kiến trúc của YOLOv2 đã được thêm vào đó những lớp BatchNorm để việc training nhanh hơn và ổn định hơn. Với việc thêm vào BatchNorm, Dropout được loại bỏ khỏi model mà không sợ bị overfitting.

High-res Classifier: Backbone của YOLOv2 được pretrained trên ImageNet. Trong YOLOv1, backbone được train trên ImageNet với kích thước ảnh 224×224 , lúc train detection với toàn bộ model thì lại sử dụng kích thước ảnh 448×448 . Việc chuyển đột ngột như vậy khiến model phải vừa học Object Detection lại còn vừa phải thích ứng với kích thước ảnh mới. Vì vậy, trong YOLOv2, backbone trước tiên được finetune trên ImageNet với kích thước ảnh 448×448 trong vòng 10 epochs, rồi mới chuyển sang dạng Object Detection.



Hình 2.3: Kiến trúc của YOLOv2 với một skip-connection

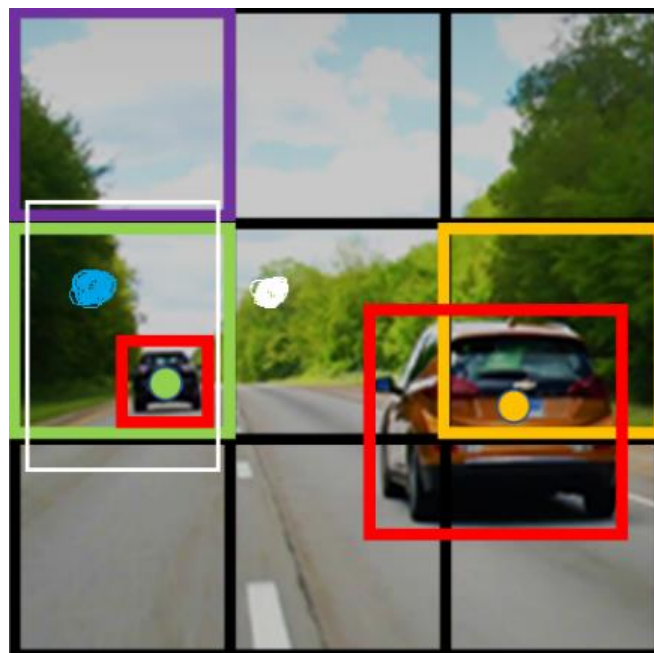
Fine-grained Features: Thay vì dự đoán trên 7×7 grid feature map như YOLOv1 thì YOLOv2 dự đoán trên một 13×13 grid feature map, việc này sẽ khiến YOLOv2 dự đoán những object nhỏ tốt hơn. Hơn nữa, YOLOv2 cũng sử dụng một skip-connection để kết hợp thông tin từ feature map ở layer trước đó vào feature map cuối (Hình 2.3).

Multi-scale training: Trước đó, YOLOv2 chỉ train với kích thước ảnh 448×448 . Sau khi áp dụng Anchor Box (sẽ trình bày ở phần sau), YOLO đổi kích thước ảnh thành 416×416 . Tuy nhiên, YOLOv2 muốn model có thể phát hiện tốt với nhiều kích thước ảnh khác nhau, vì vậy, cứ mỗi 10 batches, YOLOv2 lại thay đổi kích thước ảnh đầu vào một lần. Sở dĩ điều này có thể thực hiện được vì kiến trúc mạng của YOLOv2 hoàn toàn tạo từ các lớp Conv và có hệ số suy giảm là 32. Do đó, kích thước ảnh đầu vào của YOLOv2 thay đổi với kích thước là bội số của 32, được lấy trong khoảng $\{320, 352, \dots, 608\}$.

Ngoài ra YOLOv2 còn áp dụng thêm mới Anchor Box (hộp neo):

Anchor Box là kỹ thuật được sử dụng trong Faster R-CNN. Lúc này, kích thước ảnh đầu vào được chuyển từ 448×448 thành 416×416 vì muốn feature map thu được ở lớp cuối cùng là số lẻ (với kích thước ảnh 448×448 thì feature map ở lớp cuối là 14×14) để luôn có một ô trung tâm của feature map. Ý tưởng này đến từ việc các ảnh trong dataset COCO thường có một vật ở giữa ảnh, vì vậy, việc có một ô trung tâm của feature map để Anchor Box có thể dễ dàng lấy được luôn vật đó. Sử dụng Anchor Box, YOLOv2 bị giảm đi 0.3 mAP nhưng bù lại, Recall tăng lên. Tức là việc sử dụng Anchor Box khiến YOLOv2 phát hiện được nhiều vật thể hơn, nhưng bù lại, khả năng phát hiện chính xác lại kém đi. Trong các mô hình 2

pha (họ nhà R-CNN), việc Anchor Box hoạt động rất tốt vì pha thứ nhất đã bao gồm việc tối ưu vị trí cho Bounding Box từ Anchor Box, còn trong YOLO thì không có. Do vậy, việc có các Anchor Box đẹp được sinh ra ngay từ lúc đầu khá là quan trọng. YOLOv2 đã thêm vào bước chọn các chỉ số cho Anchor Box được sinh ra ngay từ lúc đầu thông qua thuật toán k-means. Thêm vào đó, việc tối ưu vị trí Bounding Box từ Anchor Box sinh ra được sử dụng trong Faster R-CNN đó chính là: model sẽ dự đoán độ dịch chuyển của Anchor Box gọi là t_x , t_y để suy ra vị trí của tâm Bounding Box x , y thông qua một công thức biến đổi. YOLOv2 nhận thấy việc này là không phù hợp, nên đã thêm vào một số giới hạn. YOLOv2 cũng vẫn dự đoán độ dịch chuyển t_x , t_y , t_w , t_h và Objectness Score t_o , nhưng, lúc này, t_x , t_y bị giới hạn giá trị của chúng trong khoảng $[0,1]$, việc này giới hạn vị trí tâm x , y của Bounding Box khi thực hiện phép biến đổi với t_x , t_y tức là khi dự đoán t_x , t_y tại một grid cell sẽ không khiến cho tâm x , y bị ra ngoài grid cell đó.



Hình 2.4: Khi có công thức giới hạn tâm của Bounding Box (màu xanh) sẽ nằm trong grid cell sinh ra Anchor Box và ngược lại tâm của Bounding Box (màu trắng) nằm ngoài grid cell sinh ra Anchor Box

2.1.1.3. YOLOv3

YOLOv3 đã cải tiến hơn so với YOLOv2 gồm:

- Một backbone mới: kết hợp skip-connection vào trong backbone, tăng số lớp Convolution lên.
- Thêm Feature Pyramid Network, thực hiện predict tại 3 scale

Backbone: YOLOv3 xây dựng một backbone mới, gọi là Darknet-53. Backbone của YOLOv1 thì sử dụng 1×1 Convolution (gọi là Bottleneck) của Inception Network, lên YOLOv2 thì áp dụng thêm BatchNorm, sang YOLOv3 thì áp dụng thêm skip-connection từ ResNet, gọi là một Residual Block (Hình 2.5).

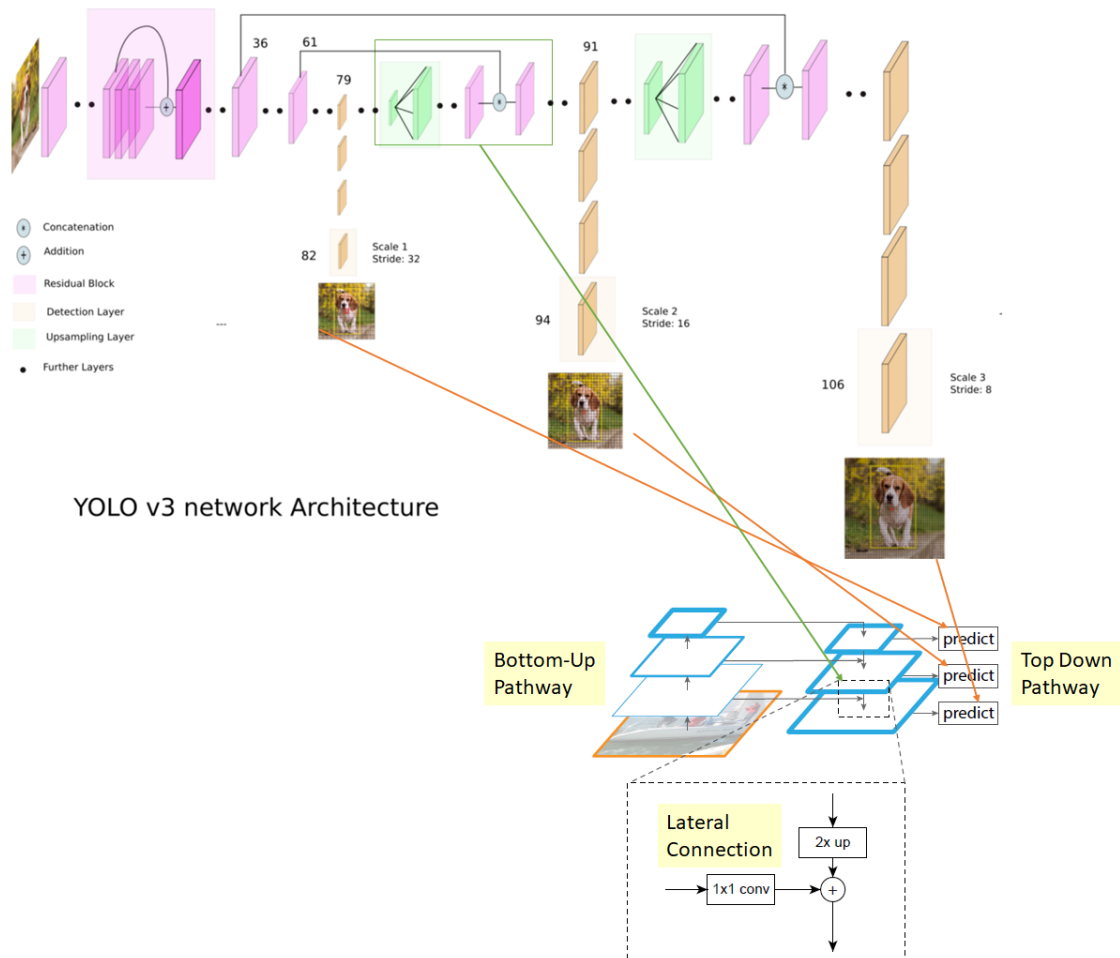
	Type	Filters	Size	Output
	Convolutional	32	3×3	256×256
	Convolutional	64	$3 \times 3 / 2$	128×128
1x	Convolutional	32	1×1	
	Convolutional	64	3×3	
	Residual			128×128
	Convolutional	128	$3 \times 3 / 2$	64×64
2x	Convolutional	64	1×1	
	Convolutional	128	3×3	
	Residual			64×64
	Convolutional	256	$3 \times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

Hình 2.5: Kiến trúc mạng backbone của YOLOv3

Neck: Từ các phiên bản YOLO trước, phát hiện vật thể nhỏ luôn là một điểm yếu. Dù trong YOLOv2 đã sử dụng skip-connection từ layer trước đó để đưa thông tin từ feature map có kích thước lớn hơn vào feature map đằng sau, nhưng điều đó là không đủ. YOLOv3 là một sự nâng cấp cho vấn đề này, áp dụng Feature Pyramid Network, thực hiện phát hiện object ở 3 scale khác nhau của feature map (Hình 2.6).

Từ các layers cấp thấp, ảnh đầu vào càng vào sâu các layers phía sau, độ phức tạp đặc trưng ngữ nghĩa sẽ tăng lên nhiều hơn trong khi độ phân giải không gian của bản đồ đặc trưng sẽ giảm xuống do kích thước giảm. Điều này dẫn đến mất các thông tin về mặt không gian cũng như các đặc trưng chi tiết. Để bảo toàn các đặc trưng chi tiết này kiến trúc Feature Pyramid Network (FPN) giúp tái tạo lại các thông tin bị mất nhờ các kết nối (skip-connection) giữa các đặc trưng (feature map) được lấy tích chập 1×1 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.

Kiến trúc FPN đã triển khai một đường dẫn từ trên xuống để chuyển các đặc trưng ngữ nghĩa (từ các layers cấp cao) và sau đó nối chúng với các đặc trưng chi tiết (từ các layers cấp thấp trong backbone) để dự đoán các đối tượng nhỏ trong với các đối tượng khác có tỷ lệ lớn hơn.



Hình 2.6: Kiến trúc của YOLOv3 với Feature Pyramid Network (FPN)

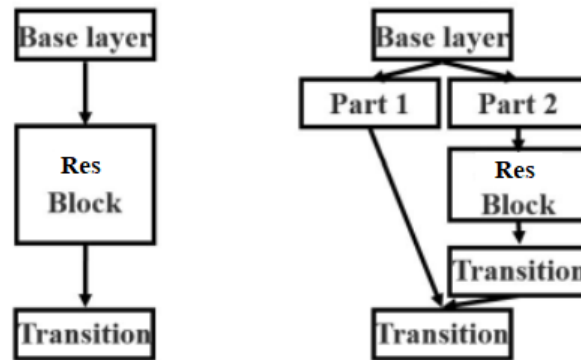
2.1.1.4. YOLOv4

Kiến trúc của YOLOv4 được nâng cấp từ YOLOv3 gồm:

- Backbone: DarkNet53 → CSPDarkNet53, áp dụng thêm DropBlock
- Neck: FPN → SPP (Spatial Pyramid Pooling) + PAN (Path Aggregation Network)
- Head: Giữ nguyên của YOLOv3

Backbone:

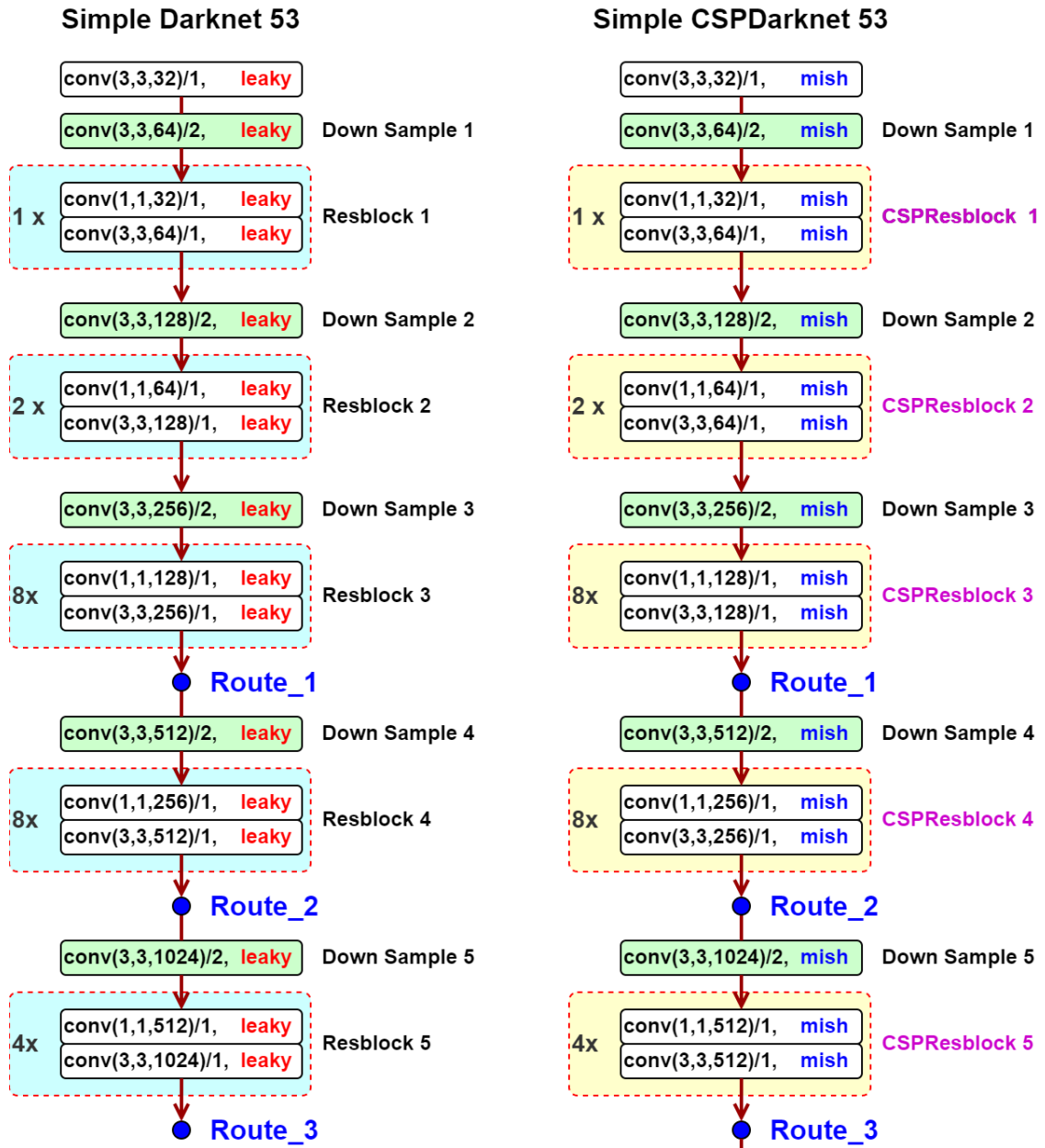
YOLOv3 sử dụng Residual Block nhưng sang YOLOv4 sử dụng ý tưởng CSPBlock



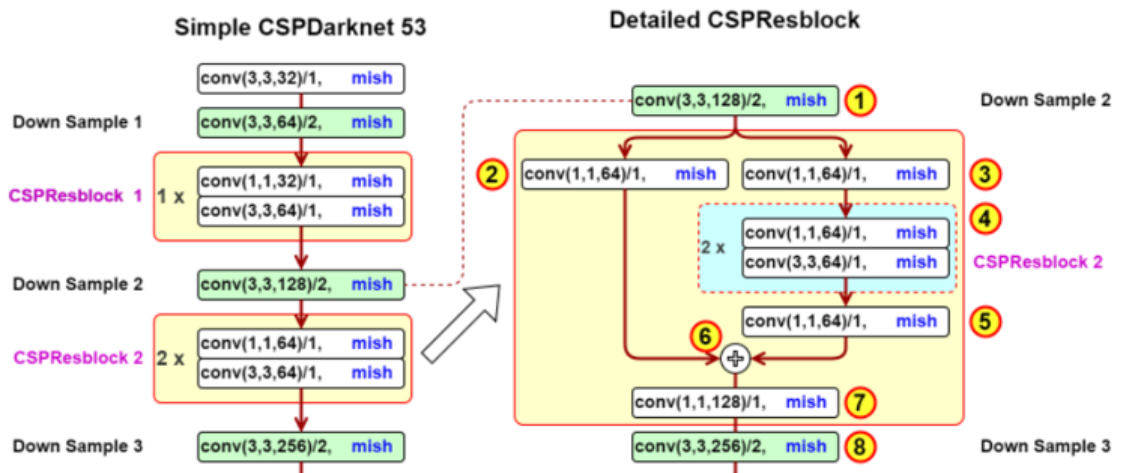
Hình 2.7: Sự khác nhau giữa Residual Block (bên trái) và CSPResBlock (bên phải)

Thay vì chỉ có một đường đi từ đầu tới cuối, CSPBlock chia thành 2 đường đi. Nhờ việc chia làm 2 đường như vậy, ta sẽ loại bỏ được việc tính toán lại gradient (đạo hàm), nhờ đó có thể tăng tốc độ trong training. Hơn nữa, việc tách làm 2 nhánh, với mỗi nhánh là một phần được lấy từ feature map trước đó, nên số lượng tham số cũng giảm đi đáng kể, từ đó tăng tốc trong cả quá trình phát hiện vật thể chứ không chỉ trong training.

YOLOv4 áp dụng ý tưởng của CSPBlock, thay thế Residual Block thông thường của YOLOv3 thành CSPResBlock, đồng thời đổi activation function từ LeakyReLU thành Mish, tạo nên CSPDarkNet53 (Hình 2.8).

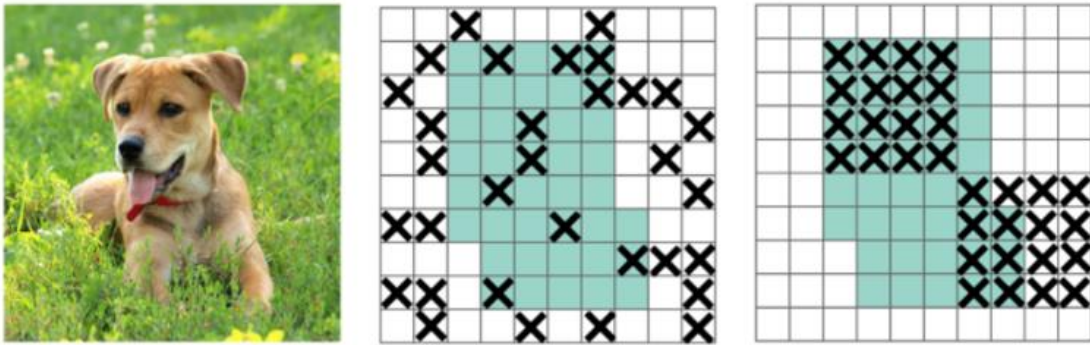


Hình 2.8: So sánh DarkNet53 với CSPDarkNet53



Hình 2.9: Chi tiết cụ thể của CSPResBlock

DropBlock: Trong các bài Classification, ở layer cuối ta thường hay sử dụng DropOut để làm giảm hiện tượng Overfitting. Nhưng trong Convolution thì việc bỏ đi random một số vị trí ở trong feature map có vẻ không hợp lý lắm. Vì các vị trí ở cạnh nhau trong feature map có tương quan cao với nhau, nên việc bỏ đi các vị trí một cách random dường như sẽ không đem lại nhiều hiệu quả. DropBlock sẽ bỏ đi nhóm vị trí trong feature map thay vì chỉ bỏ đi một vị trí (Hình 2.10).

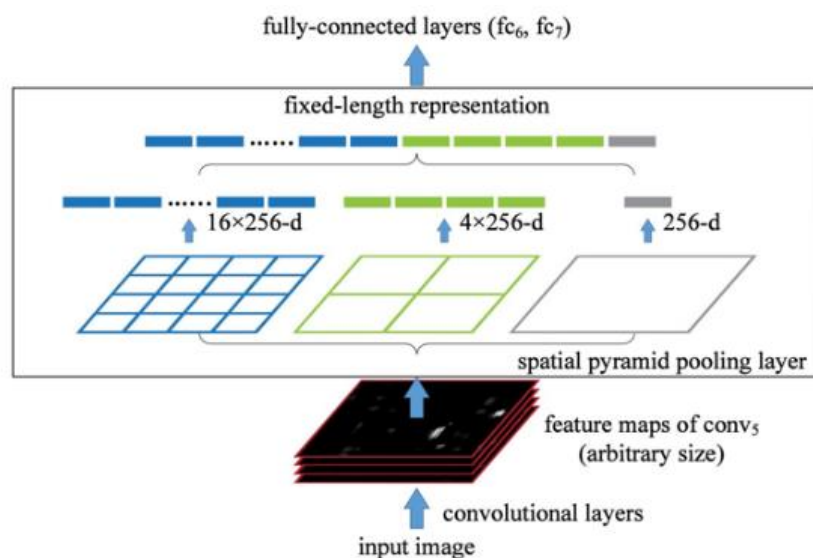


Hình 2.10: Ở giữa là DropOut, bên phải là DropBlock. Việc loại bỏ đi một nhóm các vị trí gần nhau mang lại nhiều ảnh hưởng đến lớp Convolution hơn

Neck:

YOLOv4 sử dụng 2 thành phần cho Neck là SPP và PAN.

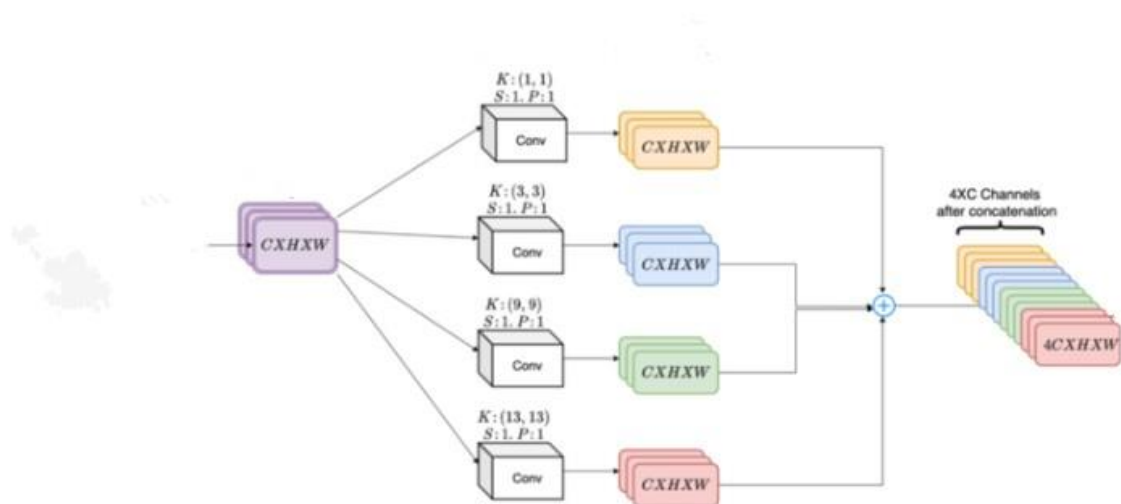
SPP(Spatial Pyramid Pooling) lần đầu được giới thiệu vào năm 2014, thời đó khi mà Object Detection vẫn còn đang sử dụng các lớp Fully-Connected ở đầu ra. Với một feature map đầu ra $C \times H \times W$ từ backbone, trước khi đưa vào lớp Fully-Connected để thực hiện detect, ta áp dụng 3 lần Max Pooling lên feature map đó như Hình 2.11.



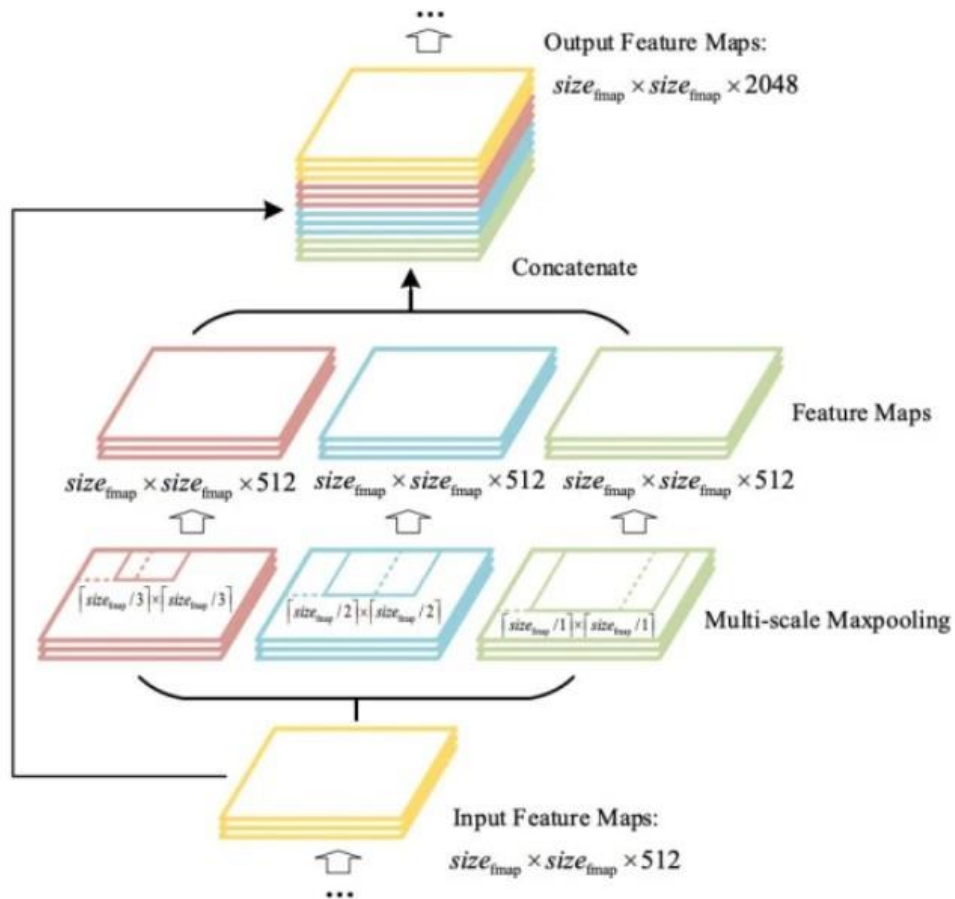
Hình 2.11: SPP ban đầu

Cách hoạt động của SPP với đầu ra fully-connected là: Mỗi kênh trong feature map từ backbone với kích thước ($H \times W \times 256$) sẽ được Max Pooling với các kernel có kích thước khác nhau, tạo ra 3 feature map với kích thước ($4 \times 4 \times 256$, $2 \times 2 \times 256$, $1 \times 1 \times 256$). Các feature map này sau đó sẽ được làm phẳng thành một vector có chiều dài ($16 \times 256 + 4 \times 256 + 256$) để đưa vào lớp Fully-connected. Thông thường, với các mạng có lớp Fully-Connected, kích thước ảnh đầu vào cần phải được giữ nguyên. Tuy nhiên, với sự xuất hiện của SPP, ảnh đầu vào đã có thể có kích thước đa dạng hơn mà không cần phải giữ nguyên nữa, một vector có độ dài cố định vẫn được sinh ra khi đi qua SPP.

YOLOv4 đã thay đổi SPP đi một chút. YOLOv4 đổi kích thước của các kernel của Max Pooling, thay vì đưa ra các feature map có kích thước 4×4 , 2×2 , 1×1 , YOLOv4 đưa ra 4 khối feature map có kích thước $H \times W$ (cùng kích thước với feature map lấy từ backbone), mỗi khối feature map $H \times W$ này được tạo ra từ Max Pooling với kernel có kích thước (1, 3, 9, 13). Sau đó, chúng được concatenate lại với nhau, tạo thành một khối feature map $H \times W \times (4 \times C)$ (Hình 2.12)



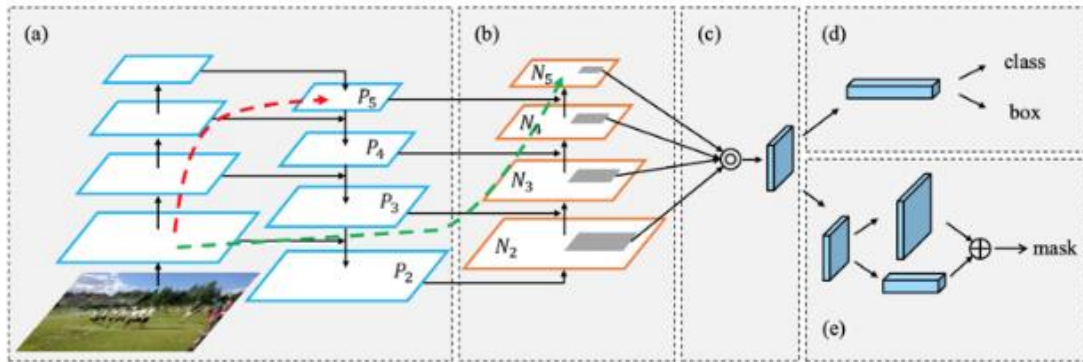
Hình 2.12: SPP trong YOLOv4. Feature map $C \times H \times W$ từ backbone được thực hiện max pooling 4 lần với các kernel có kích thước (1, 3, 9, 13), tạo ra các feature map cùng kích thước với backbone $C \times H \times W$. Các feature map này được concatenate lại với nhau thành feature map $(4 \times C) \times H \times W$



Hình 2.13: Ví dụ SPP trong YOLOv4 với $C=512$

SPP được gắn vào cuối của backbone.

PAN (Path Aggregation Network): được cải tiến từ FPN. Trong FPN, một nhánh phụ được tạo ra để đưa thông tin từ các feature ở các layer sâu vào feature ở các layer nông. Nhưng FPN lại thiếu một việc đó chính là đưa feature từ các layer nông ngược lại vào các layer sâu. Vì vậy, PAN tạo ra thêm một nhánh phụ nữa, nhánh phụ $N_2 - N_5$ được dùng để đưa thông tin ngược lại từ layer nông vào layer sâu. Thông tin từ layer nông phải đi qua cực kì nhiều layer trước khi đến layer sâu, còn trong nhánh phụ này, thông tin từ layer nông chỉ đi qua vài layer.



Hình 2.14: Mô hình mạng PAN

Head: Từ YOLOv3 trở đi head hay còn gọi là Detection Head không thay đổi và sẽ được trình bày chi tiết trong phần cấu tạo mạng YOLOv5 bên dưới.

2.1.2 Mạng YOLOv5

2.1.2.5. Cấu trúc mô hình YOLOv5

Cấu trúc YOLOv5 bao gồm năm thành phần chính:

- Bộ tiền xử lý đầu vào (Input preprocessing)
- Backbone
- Neck
- Head
- Bộ xử lý đầu ra (Output processing)

a. Bộ tiền xử lý đầu vào

Là bước đầu tiên trong quá trình xử lý hình ảnh trước khi đưa vào mô hình YOLOv5. Mục đích của bộ tiền xử lý đầu vào là chuẩn bị dữ liệu hình ảnh sao cho phù hợp với yêu cầu của mạng CNN. Các bước trong bộ tiền xử lý đầu vào bao gồm:

- Bước 1: Thay đổi kích thước hình ảnh (Image resizing): Hình ảnh đầu vào thường có kích thước và tỷ lệ khung hình (aspect ratio) khác nhau. Để đưa hình ảnh vào mô hình YOLOv5, chúng ta cần thay đổi kích thước hình ảnh sao cho phù hợp với kích thước đầu vào của mạng CNN. Điều này có thể được thực hiện bằng cách sử dụng các phương pháp nội suy như "bilinear interpolation" hoặc "bicubic interpolation".
- Bước 2: Đệm (Padding): Sau khi thay đổi kích thước hình ảnh, tỷ lệ khung hình có thể không phù hợp với kích thước đầu vào của mạng

CNN. Để giữ nguyên tỷ lệ khung hình, chúng ta có thể thêm đệm vào các cạnh của hình ảnh. Đệm thường được thêm vào dưới dạng các giá trị điểm ảnh cố định (ví dụ, 0).

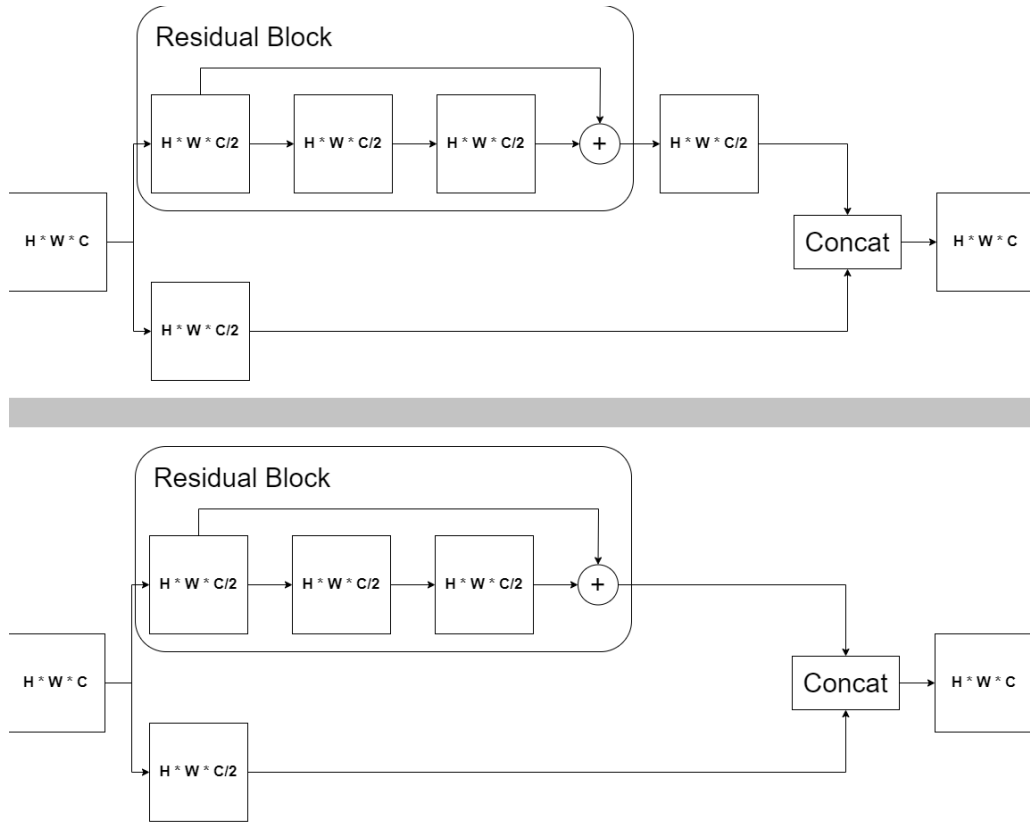
- Bước 3: Chuẩn hóa giá trị điểm ảnh (Pixel value normalization): Để giúp việc huấn luyện mô hình nhanh hơn và ổn định hơn, giá trị điểm ảnh của hình ảnh đầu vào thường được chuẩn hóa. Các giá trị điểm ảnh ban đầu trong khoảng $[0, 255]$ sẽ được chuyển đổi về khoảng $[0, 1]$ hoặc $[-1, 1]$ bằng cách chia cho 255 hoặc trừ đi 127.5 và chia cho 127.5, tùy thuộc vào yêu cầu của mô hình.
- Bước 4: Sắp xếp kênh màu (Channel ordering): Đầu vào của mạng CNN thường yêu cầu một thứ tự nhất định của kênh màu (RGB, BGR, ...). Chúng ta cần sắp xếp lại các kênh màu của hình ảnh đầu vào theo đúng thứ tự mà mô hình yêu cầu.
- Bước 5: Chuyển đổi hình ảnh thành tensor (Image to tensor conversion): Tensor là một cấu trúc dữ liệu đa chiều, giúp thao tác và xử lý dữ liệu hiệu quả trong mạng CNN. Hình ảnh, tensor thường có dạng (số lượng hình ảnh, chiều cao, chiều rộng, số kênh màu).

b. Backbone

Là phần chính của mạng CNN trong mô hình YOLOv5, nơi trích xuất đặc trưng từ hình ảnh đầu vào. Backbone đóng vai trò quan trọng trong việc mã hóa thông tin không gian và ngữ cảnh của hình ảnh để giúp mô hình dự đoán vị trí, kích thước và loại đối tượng một cách chính xác.

Trong mô hình YOLOv5, backbone sử dụng CSPNet (Cross Stage Hierarchical Networks) làm kiến trúc cơ bản. CSPNet là một kiến trúc mạng CNN hiệu quả, giúp giảm độ phức tạp tính toán và tăng tốc độ xử lý, đồng thời giữ được độ chính xác cao. CSPNet sử dụng cấu trúc phân tầng để chia các đặc trưng của hình ảnh đầu vào thành nhiều nhánh, sau đó kết hợp chúng lại ở cuối mạng để tạo ra đặc trưng phong phú và đa dạng.

Cấu trúc của CSPNet tương tự với CSPDarknet53 (hình 2.8) của YOLOv4 với cải tiến CSPResBlock thành một module mới, ít hơn một lớp Convolution gọi là C3 module.



Hình 2.15: Sự khác biệt giữa CSPResBlock trong YOLOv4 (trên) và C3 Module (dưới)

YOLOv4 sử dụng hàm phi tuyến (activation function) là Mish hoặc LeakyReLU trong khi YOLOv5 sử dụng hàm kích hoạt phi tuyến là SiLU. Hàm kích hoạt phi tuyến SiLU (sigmoid linear unit) là hàm kích hoạt phi tuyến sử dụng trong Conv module nhằm cải thiện hiệu quả trong việc phát hiện đối tượng của mạng và tối ưu hoá khả năng hội tụ của mô hình. SiLU là phiên bản cải thiện hơn và hoạt động trơn tru hơn so với hàm kích hoạt phi tuyến ReLU (rectified linear unit) thường được sử dụng. Định nghĩa về hàm SiLU được mô tả như sau:

$$ReLU = \max(0, x) \quad (PT\ 2.1)$$

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \quad (PT\ 2.2)$$

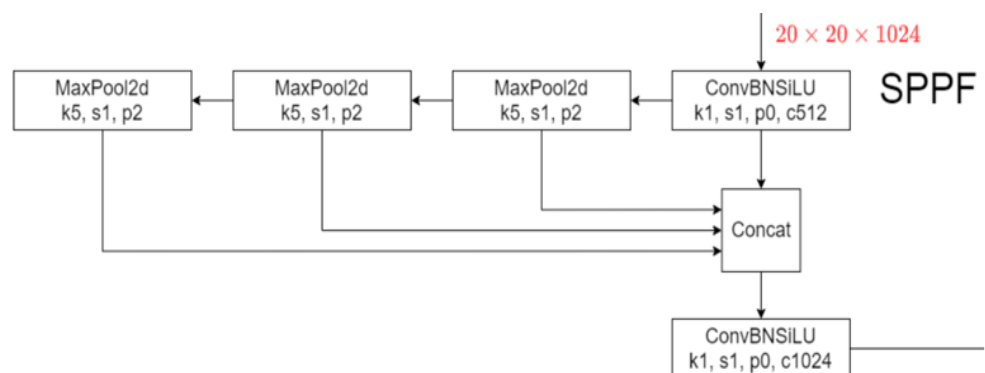
$$SiLU = x * Sigmoid(x) \quad (PT\ 2.3)$$

c. Neck

Trong mô hình YOLOv5, Neck đóng vai trò là bộ phận trung gian giữa Backbone và Head. Mục đích của Neck là kết hợp thông tin từ các đặc trưng trích xuất ở nhiều tỷ lệ khác nhau từ Backbone, tạo ra các đặc trưng đa tỷ lệ phong phú hơn để cải thiện khả năng phát hiện đối tượng ở các kích thước

và tỷ lệ khác nhau. Neck bao gồm PAN (Path Aggregation Network) và SPPF (Spatial Pyramid Pooling fast).

- PAN được thiết kế để tổng hợp thông tin từ các đặc trưng trích xuất bởi Backbone thông qua việc kết hợp đường dẫn ngắn (short path) và đường dẫn dài (long path). Điều này giúp tận dụng thông tin không gian và ngữ cảnh từ các đặc trưng ở nhiều tỷ lệ khác nhau, cải thiện độ chính xác của mô hình trong việc phát hiện đối tượng. Cấu trúc PAN của YOLOv5 tương tự như YOLOv4 đã được trình bày bên trên.
- SPPF trong YOLOv5 gần giống với SPP của YOLOv4 nhưng nhanh hơn gấp đôi với kiến trúc như hình. Thay vì sử dụng MaxPooling song song như trong SPP, SPPF của YOLOv5 sử dụng MaxPooling tuần tự. Hơn nữa, kernel size trong MaxPooling của SPPF toàn bộ là 5 thay vì là [5,9,13] như SPP của YOLOv4.



Hình 2.16: SPPF của YOLOv5

d. Head

Head trong mô hình YOLOv5, còn được gọi là Detection Head, là phần cuối cùng của mô hình, nơi dự đoán vị trí, kích thước và loại đối tượng từ các đặc trưng đa tỷ lệ được kết hợp từ Neck. Head chịu trách nhiệm xuất ra các thông số của bounding box (hộp giới hạn) và xác suất của lớp đối tượng (object class probabilities) cho mỗi ô lưới (grid cell) trên hình ảnh.

Cấu trúc của head bao gồm:

- Lớp chập (Convolutional layers): Các lớp chập trong Head nhằm tinh chỉnh và xử lý các đặc trưng đa tỷ lệ từ Neck. Các lớp chập này

giúp chuẩn bị thông tin cho việc dự đoán vị trí, kích thước và loại đối tượng.

- Lớp dự đoán (Prediction layer): Lớp này có nhiệm vụ dự đoán các thông số của bounding box, bao gồm tọa độ trung tâm (x, y), chiều rộng và chiều cao của bounding box, cũng như xác suất của lớp đối tượng cho mỗi ô lưới. Số lượng kênh đầu ra của lớp dự đoán phụ thuộc vào số lượng đối tượng và anchor box (hộp neo) được sử dụng trong mô hình.

e. Bộ xử lý đầu ra

Bộ xử lý đầu ra (Output processing) trong mô hình YOLOv5 là giai đoạn cuối cùng của quá trình phát hiện đối tượng, nơi mô hình xử lý các dự đoán của bounding box và xác suất lớp đối tượng từ Head để trả về kết quả cuối cùng. Bộ xử lý đầu ra thực hiện các bước sau:

- Xử lý anchor box: Đưa ra dựa trên các dự đoán của Head, mô hình sử dụng anchor box (hộp neo) đã được định nghĩa trước để điều chỉnh kích thước và vị trí của các bounding box dự đoán.
- Giải mã bounding box: Chuyển đổi các dự đoán của Head về tọa độ trung tâm (x, y), chiều rộng và chiều cao của bounding box thành tọa độ góc trên bên trái ($xmin, ymin$) và góc dưới bên phải ($xmax, ymax$) của bounding box.
- NMS (Non-Maximum Suppression): Loại bỏ các bounding box trùng lặp và giữ lại các bounding box chính xác nhất. NMS giúp giảm bớt số lượng bounding box dư thừa và cải thiện độ chính xác của kết quả phát hiện.
- Xác định lớp đối tượng: Dựa trên xác suất của lớp đối tượng từ Head, mô hình xác định lớp của mỗi đối tượng được phát hiện trong bounding box.

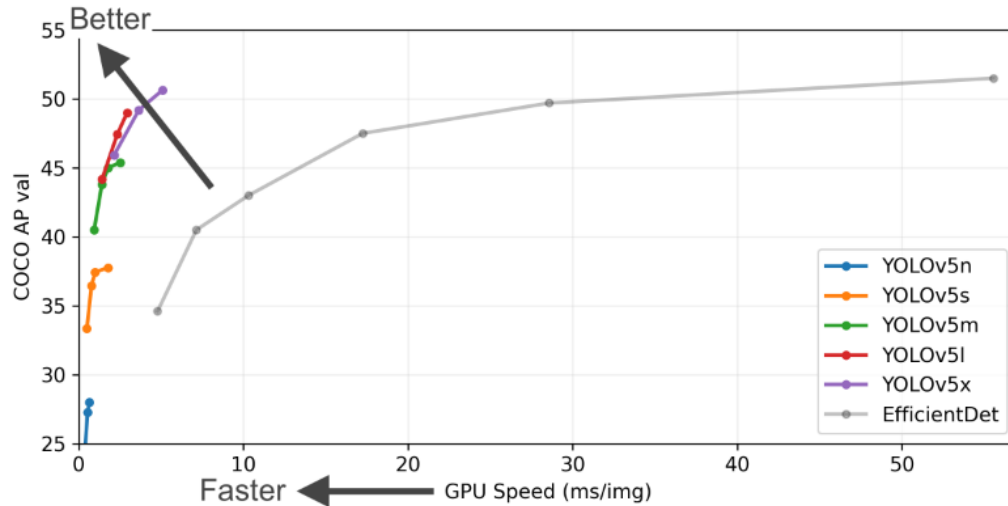
Ưu điểm của YOLOv5 so với các mô hình trước là:

- YOLOv5 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 giảm thời gian huấn luyện một cách tối ưu.

- Về hiệu năng: YOLOv5 nhỏ hơn và thường dễ sử dụng hơn trong sản xuất. Do nó được triển khai nguyên bản trong PyTorch (chứ không phải Darknet), việc sửa đổi kiến trúc và rất đơn giản trong việc xuất sang nhiều môi trường triển khai khác nhau.
- Về tốc độ : YOLOv5 thực hiện suy luận hàng loạt ở khoảng 140 FPS theo mặc định.
- Về độ chính xác: YOLOv5 gần như chính xác như YOLOv4 trong các tác vụ nhỏ (0,895 mAP so với 0,892 mAP trên BCCD).

2.1.2.6. Các biến thể của mô hình YOLO5

YOLOv5 có nhiều biến thể với kích thước và độ phức tạp khác nhau, ví dụ như YOLOv5s, YOLOv5m, YOLOv5l và YOLOv5x. Các biến thể này được thiết kế để cân bằng giữa độ chính xác và tốc độ xử lý, giúp chúng phù hợp với nhiều ứng dụng và thiết bị khác nhau. YOLOv5s là phiên bản nhỏ nhất và nhanh nhất, trong khi YOLOv5x là phiên bản lớn nhất và chính xác nhất, nhưng cũng đòi hỏi nhiều tài nguyên hơn để xử lý.



Hình 2.17 Hình ảnh so sánh các mô hình YOLOv5

2.2 Phân loại tế bào bạch cầu sử dụng mô hình kết hợp VGG16 và YOLOv5

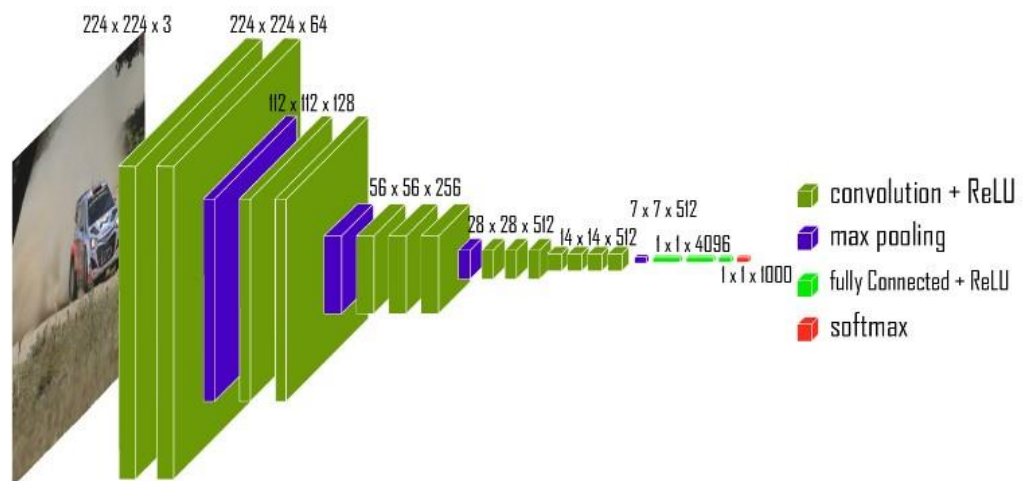
2.2.1 Mô hình mạng VGG16

VGG16 là một kiến trúc mạng nơ-ron tích chập (Convolutional Neural Network - CNN) nổi tiếng được giới thiệu bởi nhóm nghiên cứu Visual Geometry Group (VGG) thuộc Đại học Oxford trong bài báo "Very Deep Convolutional Networks for Large-Scale Image Recognition" năm 2014.

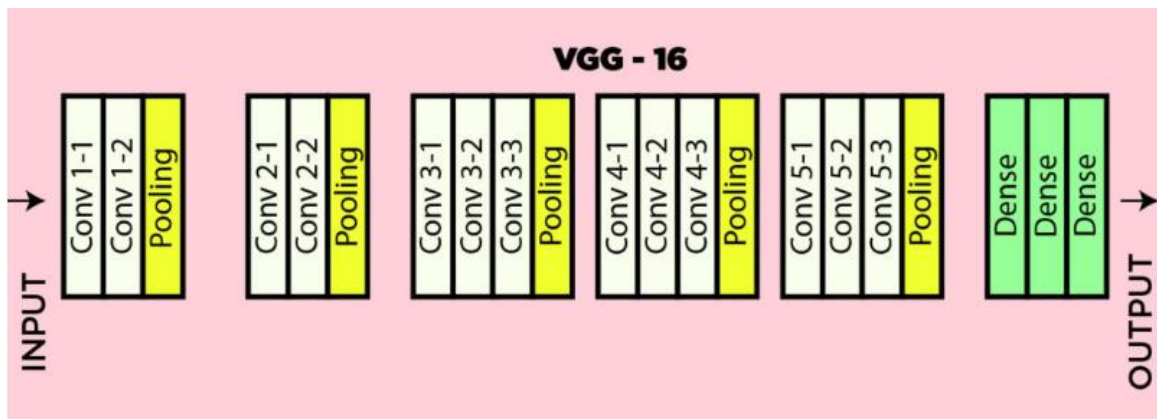
VGG16 được thiết kế với mục đích xây dựng một mạng nơ-ron tích chập sâu hơn, đơn giản hơn và hiệu quả hơn so với các kiến trúc trước đó như AlexNet. VGG16 có 16 tầng trọng số, bao gồm 13 tầng tích chập và 3 tầng kết nối đầy đủ (fully connected layers). Mô hình này sử dụng các bộ lọc kích thước 3x3, bước chập (stride) bằng 1 và đệm (padding) bằng 1 trong các tầng tích chập, cùng với các tầng gộp cực đại (max-pooling) kích thước 2x2 và bước gộp bằng 2.

Cấu trúc của VGG16 gồm các phần sau:

- 13 tầng tích chập (Convolutional layers): Mỗi tầng sử dụng bộ lọc kích thước 3x3, bước chập bằng 1 và đệm bằng 1. Số lượng bộ lọc trong mỗi tầng tăng dần từ 64, 128, 256 đến 512.
- 5 tầng gộp cực đại (Max-pooling layers): Sau mỗi nhóm các tầng tích chập, một tầng gộp cực đại được áp dụng để giảm kích thước không gian của các đặc trưng.
- 3 tầng kết nối đầy đủ (Fully connected layers): Tầng đầu tiên và thứ hai có 4096 nơ-ron, trong khi tầng cuối cùng có số lượng nơ-ron tương ứng với số lượng lớp đối tượng cần phân loại.
- Tầng Softmax: Tầng này chuyển đổi đầu ra của tầng kết nối đầy đủ cuối cùng thành các xác suất cho mỗi lớp đối tượng.



Hình 2.18: Kiến trúc của VGG16



Hình 2.19: Bản đồ kiến trúc VGG16

Chức năng của VGG16:

- Trích xuất đặc trưng: VGG16 giúp trích xuất các đặc trưng từ hình ảnh đầu vào. Điều này được thực hiện thông qua các tầng tích chập của mạng, bắt đầu từ việc nhận diện các đặc trưng cơ bản như cạnh và góc, và tiến dần đến các đặc trưng phức tạp hơn.
- Phân loại hình ảnh: VGG16 được sử dụng để phân loại hình ảnh thành các lớp đối tượng khác nhau. Điều này được thực hiện thông qua các lớp kết nối đầy đủ (FC) và lớp softmax ở cuối mạng, cho phép đưa ra xác suất cho từng lớp đối tượng.
- Đào tạo và học chuyển giao (Transfer Learning): VGG16 thường được sử dụng trong các tác vụ học chuyển giao, nơi một mạng đã được đào tạo trước đó trên một tập dữ liệu lớn (như ImageNet) được sử dụng làm điểm khởi đầu cho việc đào tạo trên một tập dữ liệu mới. Điều này giúp giảm thời gian đào tạo và cải thiện hiệu suất của mô hình trên các tập dữ liệu có ít mẫu hơn.
- Rút trích đặc trưng: Ngoài phân loại hình ảnh, VGG16 cũng có thể được sử dụng để rút trích đặc trưng từ hình ảnh, tạo ra các vector đặc trưng phục vụ cho các tác vụ khác, chẳng hạn như phân cụm hoặc tìm kiếm hình ảnh tương tự.
- Nền tảng cho các mô hình phức tạp hơn: VGG16 cũng thường được sử dụng làm nền tảng để xây dựng các mô hình phức tạp hơn, như các mô hình phân đoạn hình ảnh, phát hiện đối tượng và các tác vụ thị giác máy tính khác.

Ưu điểm của VGG16:

- Kiến trúc đơn giản: VGG16 có kiến trúc đơn giản và dễ hiểu, với các lớp tích chập và lớp gộp cực đại được sắp xếp theo cấu trúc đồng nhất. Điều này giúp cho việc lập trình và điều chỉnh mạng dễ dàng hơn.
- Hiệu suất cao: VGG16 đã đạt được kết quả tốt trên các bộ dữ liệu phân loại hình ảnh lớn, chẳng hạn như ImageNet. Mô hình này được coi là một trong những mạng CNN hiệu quả nhất cho các tác vụ phân loại hình ảnh.
- Học chuyển giao (Transfer Learning) hiệu quả: VGG16 thường được sử dụng trong các tác vụ học chuyển giao, nơi một mạng đã được đào tạo trước đó trên một tập dữ liệu lớn được sử dụng làm điểm khởi đầu cho việc đào tạo trên một tập dữ liệu mới. Điều này giúp giảm thời gian đào tạo và cải thiện hiệu suất của mô hình trên các tập dữ liệu có ít mẫu hơn.

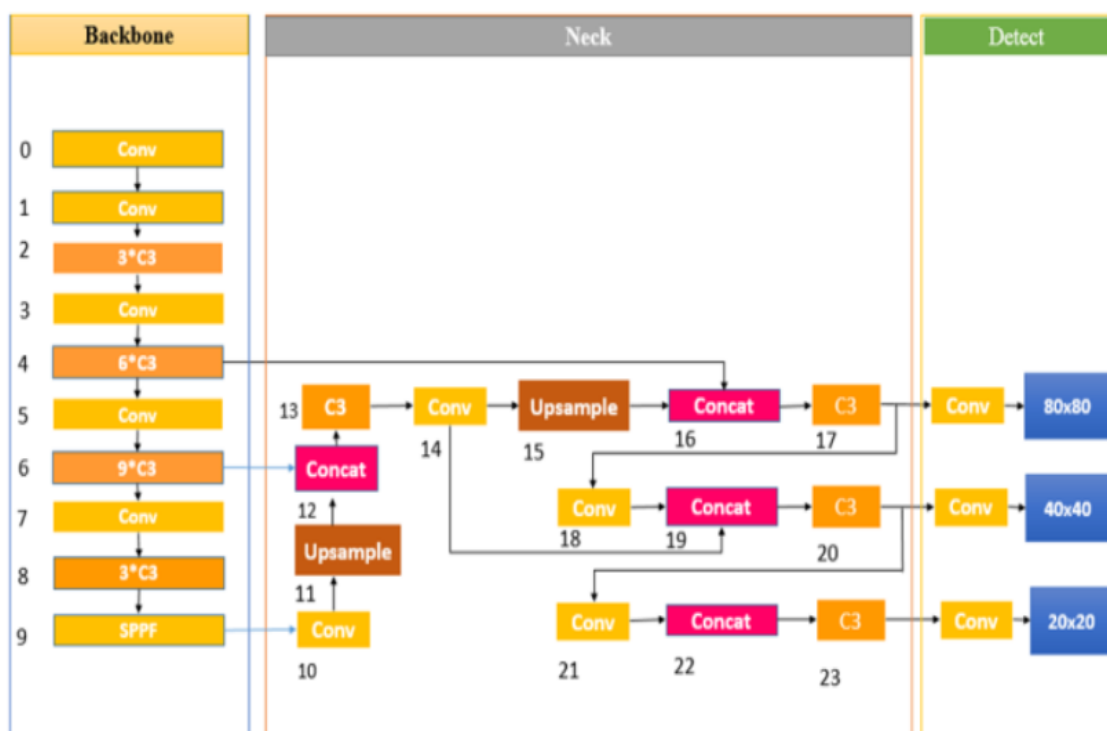
Nhược điểm của VGG16:

- Kích thước mô hình lớn: VGG16 có kích thước mô hình lớn, với khoảng 138 triệu tham số. Điều này dẫn đến việc sử dụng nhiều bộ nhớ và tài nguyên tính toán, làm chậm quá trình đào tạo và suy luận.
- Thời gian đào tạo lâu: Do kích thước mô hình lớn và số lượng tham số nhiều, VGG16 mất nhiều thời gian để đào tạo so với một số mạng CNN nhẹ nhàng hơn, chẳng hạn như MobileNet.
- Không tối ưu cho các thiết bị di động và nhúng: VGG16 không phải là lựa chọn tối ưu cho các thiết bị di động và nhúng do kích thước mô hình lớn và yêu cầu tính toán cao. Các mạng CNN nhẹ nhàng hơn, chẳng hạn như MobileNet và SqueezeNet, thường được ưu tiên hơn cho các tác vụ trên các thiết bị có tài nguyên hạn chế.
- Không cải tiến đáng kể so với các kiến trúc hiện đại hơn: Mặc dù VGG16 có hiệu suất tốt trong quá khứ, hiện nay đã có các kiến trúc CNN tiên tiến hơn, chẳng hạn như ResNet, DenseNet và EfficientNet, đạt được hiệu suất tốt hơn trong nhiều tác vụ thị giác máy tính và cũng hiệu quả hơn về mặt tính toán.

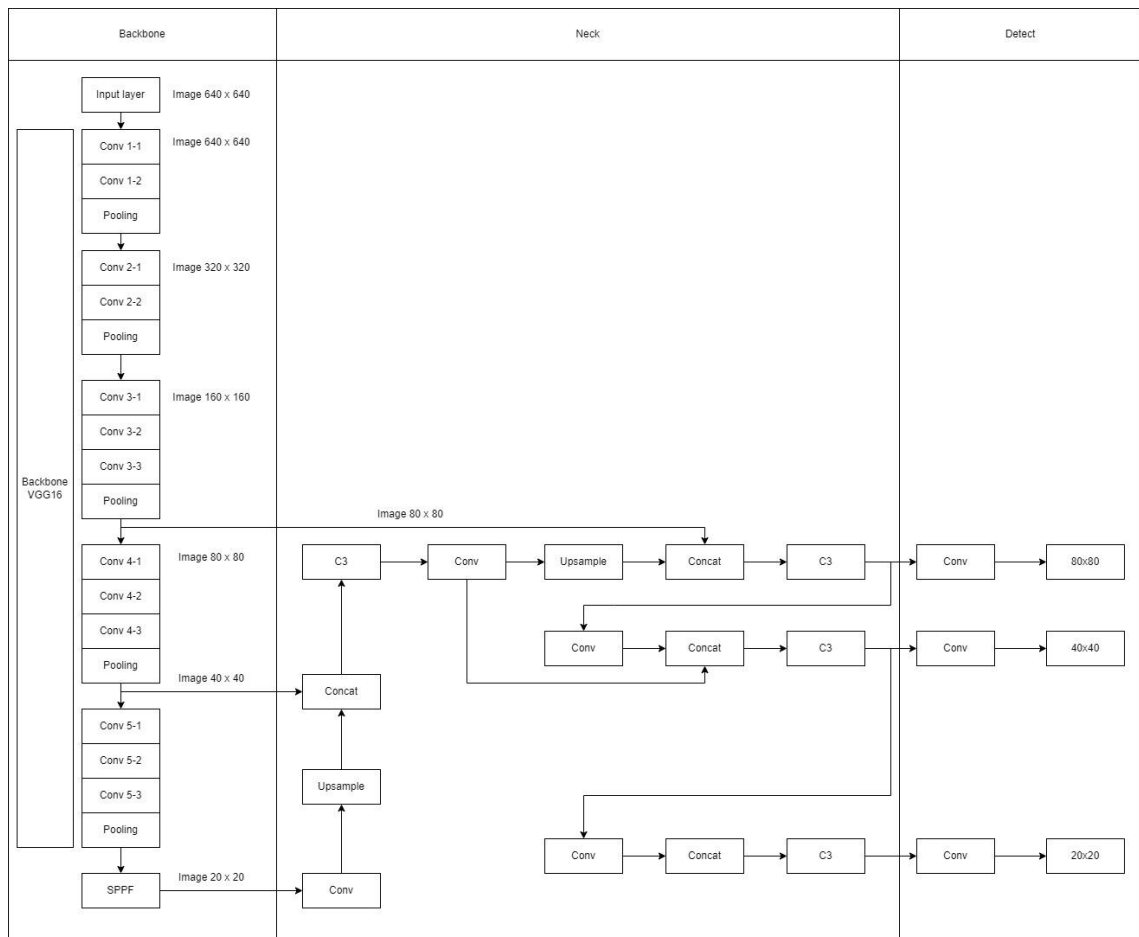
2.2.2 Mô hình kết hợp VGG16 và YOLOv5

YOLOv5 là mô hình phát hiện đối tượng mạnh mẽ đáng để nghiên cứu sâu về cấu trúc và hiệu suất của nó. Cách tốt nhất để sửa đổi đó là thử thay đổi cấu trúc của nó và thử nghiệm tác dụng.

Do đó, mô hình kết hợp VGG16 và YOLOv5 là mô hình thay thế toàn bộ backbone của YOLOv5 bằng VGG16 đảm bảo các kích thước ảnh đầu ra của backbone VGG16 tương ứng với kích thước ảnh đầu vào của phần Neck của YOLOv5 để kiểm thử mô hình với mục đích nâng cao hiệu suất, tăng độ chính xác và giảm thời gian huấn luyện mô hình.



Hình 2.20: Kiến trúc mô hình YOLOv5



Hình 2.21: Kiến trúc mô hình YOLOv5 backbone VGG16

2.3 Kết luận chương

Trong chương này, chúng ta đã đi qua hai mô hình phân loại tế bào máu. Đầu tiên, chúng ta đã tìm hiểu sơ lược về mô hình YOLO, một mô hình phát hiện đối tượng nhanh và hiệu quả, và tiếp tục đề cập tới phiên bản mới nhất của YOLO là YOLOv5.

Sau đó, chúng ta đã khám phá mô hình kết hợp VGG16 và YOLOv5. VGG16, một kiến trúc CNN nổi tiếng, được đưa vào để trích xuất đặc trưng từ hình ảnh của tế bào máu, trong khi YOLOv5 giúp phát hiện và phân loại các tế bào bạch cầu dựa trên đặc trưng này.

Chương tiếp theo sẽ tập trung vào việc thu thập dữ liệu, huấn luyện các mô hình, đánh giá hiệu quả của từng mô hình và so sánh hai mô hình để xem mô hình nào phù hợp nhất cho việc phân loại tế bào máu.

CHƯƠNG 3. KẾT QUẢ VÀ BÀN LUẬN

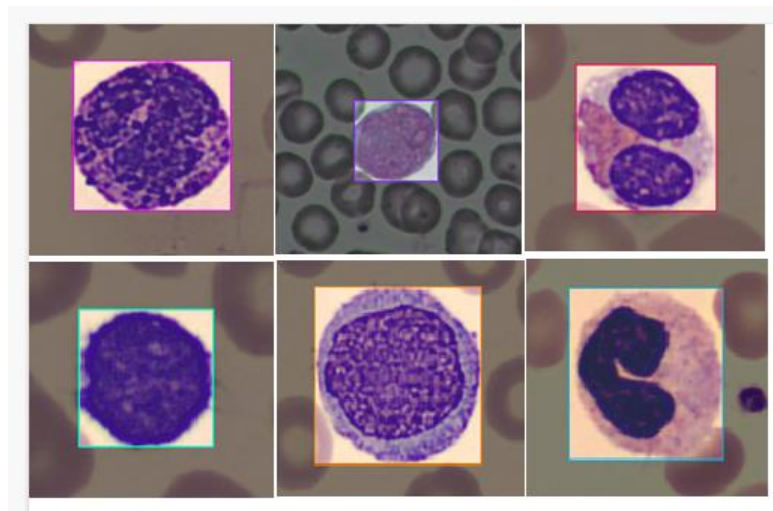
3.1 Cơ sở dữ liệu huấn luyện

3.1.1 Mô tả dữ liệu

Tác giả thu thập hình ảnh hiển vi chụp tiêu bản máu ngoại vi từ Viện Huyết Học Truyền Máu Trung Ương và Bệnh viện Đại Học Y Hà Nội. Và để tăng số lượng hình ảnh và duy trì sự đồng nhất của các tế bào bệnh lý, tác giả chỉ thu thập thêm ảnh các tế bào không phải bệnh lý từ một số bộ dữ liệu hiện được biết đến như BCCD. Hình ảnh hiển vi bạch cầu được nhuộm màu bằng phương pháp Reiter-Giemsa.

Với mục đích hỗ trợ các bác sĩ, kỹ thuật viên, đưa ra các giá trị tham khảo trong đọc lam tế bào, các bệnh viện tuyến dưới khi không có đầy đủ đội ngũ BS, KTV tay nghề cao cần phát hiện tế bào bất thường để sớm đưa bệnh nhân chuyển lên tuyến trên điều trị và đồng thời cũng là tài liệu tham khảo để giảng dạy cho các sinh viên do đó bài luận văn sẽ phân loại đủ 5 loại của tế bào bạch cầu (bạch cầu ái kiềm (basophil), bạch cầu ái toan (eosinophils), bạch cầu lympho (lymphocytes), bạch cầu đơn nhân (monocytes), bạch cầu đa nhân trung tính (neutrophils) và thêm 1 lớp nữa là các tế bào bạch cầu bất thường (diseased). Các đặc điểm nhân của bạch cầu nhuộm màu là tương đối rõ ràng.

Tổng bộ dữ liệu ban đầu là 1974 ảnh. Bộ dữ liệu ảnh hiển vi chụp tiêu bản máu ngoại vi được gán nhãn bằng việc sử dụng công cụ trang web hỗ trợ roboflow dưới sự tham vấn của BS, KTV giàu kinh nghiệm tại cơ sở thu nhận dữ liệu. Sáu loại bạch cầu được gán nhãn như trong Hình 3.1. Sau khi được gán nhãn, dữ liệu được xuất ra tập tin với định dạng dữ liệu của YOLOv5 pytorch. Do với số lượng ảnh chưa được 2000 ảnh chưa đủ nhiều để huấn luyện các mô hình học sâu (deep learning) vì thế cần áp dụng biện pháp tăng cường dữ liệu để có một bộ dữ liệu mới mạnh mẽ hơn để sử dụng huấn luyện mô hình.



Hình 3.1 Ảnh gán nhãn các tế bào lần lượt theo thứ tự là *basophil*, *diseased*, *eosinophil*, *lymphocyte*, *monocyte*, *neutrophil*

3.1.2 Tăng cường dữ liệu

Tăng cường dữ liệu huấn luyện (data augmentation) là một kỹ thuật được sử dụng để mở rộng và làm giàu tập dữ liệu huấn luyện bằng cách tạo ra các biến thể mới của các mẫu dữ liệu hiện có. Việc tăng cường dữ liệu huấn luyện có những mục đích chính sau:

- Ngăn ngừa overfitting: Khi một mô hình học quá tốt trên tập dữ liệu huấn luyện nhưng lại không tổng quát hóa tốt trên dữ liệu mới, điều này được gọi là overfitting. Tăng cường dữ liệu giúp tạo ra các mẫu dữ liệu mới và đa dạng hơn, giúp mô hình học được các đặc trưng tổng quát hơn và ngăn ngừa overfitting.
- Cải thiện hiệu suất: Tăng cường dữ liệu giúp mô hình học được các đặc trưng quan trọng từ nhiều biến thể của cùng một mẫu dữ liệu, dẫn đến hiệu suất tốt hơn khi gặp dữ liệu chưa từng thấy trước đây.
- Giảm thiểu thiên vị: Một tập dữ liệu huấn luyện có thể chứa thiên vị về đặc trưng, lớp hoặc phân phối. Tăng cường dữ liệu giúp làm giàu tập dữ liệu huấn luyện và giảm thiểu thiên vị.
- Xử lý dữ liệu không cân bằng: Trong trường hợp dữ liệu huấn luyện không cân bằng giữa các lớp, việc tăng cường dữ liệu giúp cải thiện số lượng mẫu của các lớp ít biểu diễn, giúp mô hình học tốt hơn trên các lớp này.

Một số kỹ thuật tăng cường dữ liệu phổ biến bao gồm: lật ảnh (flipping), xoay ảnh (rotation), dịch chuyển (translation), cắt xén (cropping), thay đổi độ

sáng/độ tương phản (brightness/contrast adjustment), và thêm nhiễu (noise addition).

Trong bài viết này, chúng ta dùng kỹ thuật lật ảnh (horizontal flip), xoay ảnh (rotation) và thay đổi độ sáng/độ tương phản ((brightness/contrast adjustment).

3.1.2.1. *Kỹ thuật lật ảnh ((horizontal flip)*

Horizontal flip (lật ngang) là một kỹ thuật tăng cường dữ liệu được sử dụng trong xử lý ảnh và thị giác máy tính. Khi áp dụng kỹ thuật này, một bức ảnh sẽ được lật theo trục ngang, nghĩa là mọi điểm ảnh ở phía bên trái của ảnh gốc sẽ chuyển sang phía bên phải của ảnh mới, và ngược lại. Điều này dẫn đến việc tạo ra một bức ảnh đối xứng với bức ảnh gốc thông qua trục ngang.

Horizontal flip giúp mở rộng tập dữ liệu huấn luyện bằng cách tạo ra các biến thể mới của ảnh gốc, giúp mô hình học được các đặc trưng của đối tượng dù chúng xuất hiện ở vị trí đối xứng.

3.1.2.2. *Xoay ảnh (rotation)*

Khi áp dụng kỹ thuật này, một bức ảnh sẽ được xoay một góc 90 độ theo chiều kim đồng hồ hoặc ngược chiều kim đồng hồ.

Mục đích của việc xoay ảnh 90 độ là tạo ra các biến thể mới của ảnh gốc, giúp mô hình học được các đặc trưng của đối tượng dù chúng có bị xoay. Điều này giúp mô hình thị giác máy tính có khả năng nhận diện và phân loại đối tượng trong các tình huống đa dạng hơn, bất kể đối tượng có bị xoay hay không.

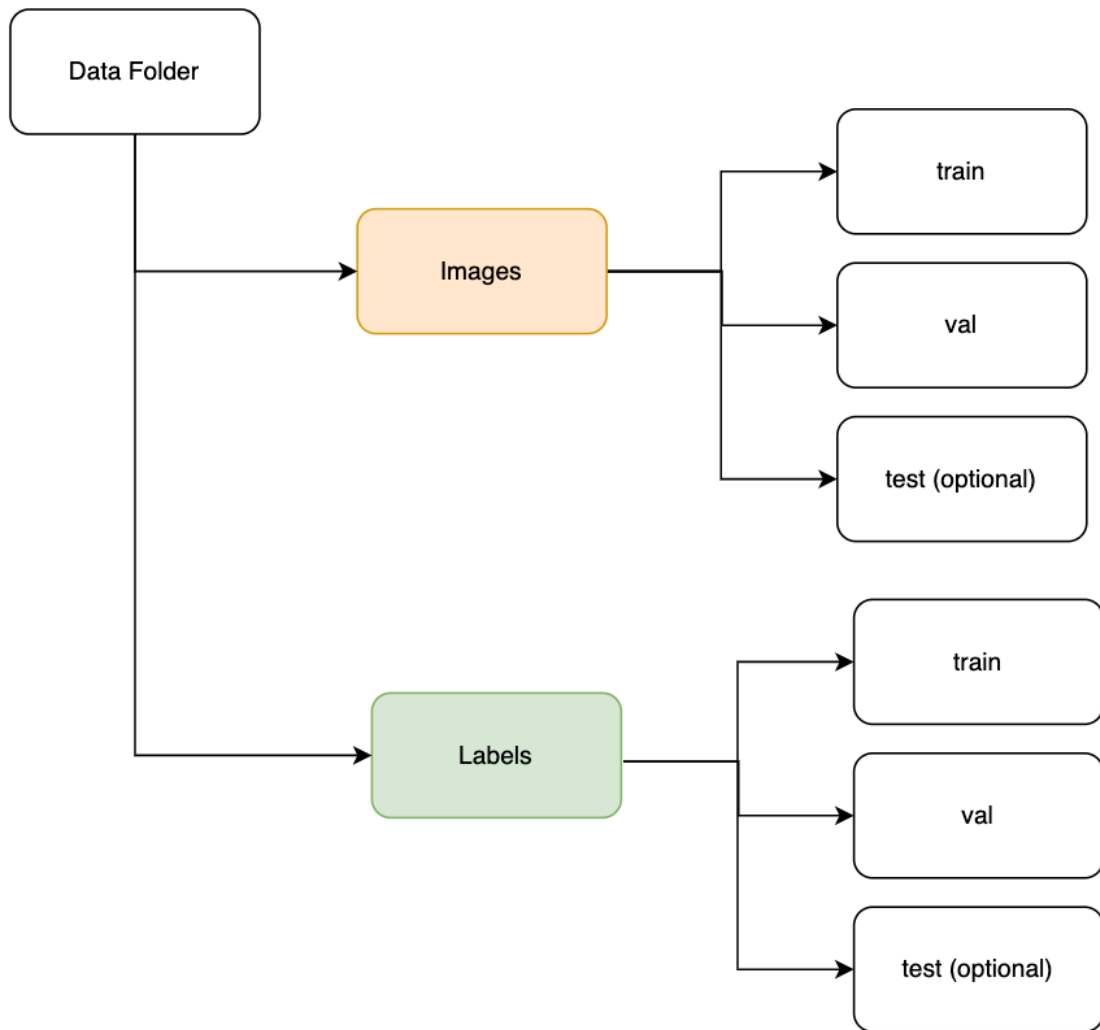
3.1.2.3. *Thay đổi độ sáng/độ tương phản*

Mục đích của việc thay đổi độ sáng và độ tương phản là để tạo ra các biến thể mới của ảnh gốc, giúp mô hình học được các đặc trưng của đối tượng trong điều kiện chiếu sáng và tương phản khác nhau.

Việc thay đổi độ sáng và độ tương phản của ảnh có thể được thực hiện bằng nhiều cách, nhưng một phương pháp phổ biến là sử dụng các phép biến đổi tuyến tính trên các giá trị màu sắc của ảnh. Ví dụ, để thay đổi độ sáng của ảnh, chúng ta có thể cộng (hoặc trừ) một hằng số vào từng giá trị màu sắc của ảnh. Để thay đổi độ tương phản, chúng ta có thể nhân (hoặc chia) từng giá trị màu sắc của ảnh với một hằng số. Bằng cách kết hợp cả hai phép biến đổi này, tạo ra các biến thể của ảnh gốc với độ sáng và độ tương phản khác nhau.

3.1.3 Dữ liệu huấn luyện

Sau khi tăng cường cơ sở dữ liệu ta có tổng cộng 6532 ảnh với số lớp cần phân loại là 6 lớp (basophil, diseased, eosinophil, lymphocyte, monocyte, neutrophil). Sau đó ta chia dữ liệu thành 2 tập chính là tập train và valid với tỉ lệ tương ứng là 90% và 10%



Hình 3.2 Sơ đồ chia dữ liệu huấn luyện mô hình

3.2 Công cụ lập trình

Huấn luyện mô hình kết hợp giữa máy tính PC cấu hình i5-10400F CPU @ 2.90GHz 2.90 GHz, 16GB Ram, GPU GTX750Ti và google colab với cấu hình 40GB Ram, GPU NVIDIA A100-SXM4-40GBm.

Google Colab (viết tắt của "Google Colaboratory") là một môi trường lập trình Jupyter Notebook được cung cấp miễn phí bởi Google. Nó cho phép người dùng

tạo, chia sẻ và chỉnh sửa các tập lệnh Python trực tiếp trong trình duyệt web mà không cần cài đặt phần mềm hay thiết lập cấu hình đặc biệt nào.

Google Colab hỗ trợ tích hợp nhiều thư viện mã nguồn mở phổ biến dành cho Python, bao gồm các thư viện học máy và học sâu như TensorFlow, PyTorch, Keras và scikit-learn. Nó cũng cung cấp các tài nguyên tính toán miễn phí, bao gồm CPU, GPU và TPU, giúp người dùng đào tạo và thử nghiệm các mô hình học sâu mà không cần phải sở hữu phần cứng mạnh mẽ.

Ngôn ngữ lập trình ở đây là python cùng framwork của pytorch.

PyTorch là một thư viện mã nguồn mở phổ biến dành cho Python, được phát triển bởi Facebook's AI Research Lab (FAIR). PyTorch cung cấp một nền tảng linh hoạt và hiệu suất cao để xây dựng và đào tạo các mô hình học sâu (deep learning). Nó được thiết kế với mục đích giúp nghiên cứu và phát triển dễ dàng hơn trong lĩnh vực học máy (machine learning) và học sâu.

3.3 Chỉ số đánh giá mô hình

Trong AI, khi chúng ta huấn luyện một mô hình, thì chúng ta cũng phải đặt ra cho mình tiêu chí đo lường khi nào dừng việc huấn luyện? Với mô hình đó thì chúng ta đã đạt được yêu cầu đề ra chưa? Đo lường qua tiêu chí gì?

Với một mô hình phân loại chó/ mèo đơn giản, thì chúng ta sẽ đo lường thông qua chỉ số precision, recall, confusion metric, accuracy...

Còn với một mô hình phát hiện và phân loại vật thể (Object detection) thì sẽ sử dụng các tiêu chí IoU, Precision, Recall, Precision Recall Curve, AUC,AP và mAP.

3.3.1 Intersection over Union

IoU viết tắt của Intersection over Union, dùng để đo lường mức độ chồng chéo giữa hai hình chữ nhật (bounding boxes) bằng cách tính tỷ lệ diện tích của phần giao nhau (intersection) và phần hợp nhau (union) của chúng.

Công thức tính IoU như sau:

$$IoU = \frac{\text{Diện tích giao nhau}}{\text{Diện tích hợp nhau}} \quad (\text{PT 3.1})$$

Giá trị của IoU nằm trong khoảng từ 0 đến 1. Khi hai bounding boxes không chồng chéo nhau, IoU bằng 0. Khi hai bounding boxes trùng khớp hoàn toàn, IoU bằng 1.

IoU thường được sử dụng để đánh giá mức độ chính xác của các dự đoán so với ground truth (nhãn chính xác từ người gán nhãn). Để xác định liệu một dự đoán có được coi là đúng hay không, người ta thường đặt một ngưỡng IoU (ví dụ, 0.5). Nếu giá trị IoU giữa dự đoán và ground truth lớn hơn hoặc bằng ngưỡng này, dự đoán được coi là chính xác; nếu không, dự đoán được coi là không chính xác.

3.3.2 Precision và Recall

Precision (độ chính xác) là một chỉ số đánh giá hiệu suất của mô hình phân loại trong lĩnh vực học máy. Precision đo lường tỷ lệ giữa số lượng dự đoán đúng và tổng số lượng dự đoán thuộc một lớp cụ thể.

Để hiểu rõ hơn về precision, ta cần nắm vững các khái niệm sau:

- True Positive (TP): Số lượng các mẫu được dự đoán là thuộc lớp đó và thực sự thuộc lớp đó.
- False Positive (FP): Số lượng các mẫu được dự đoán là thuộc lớp đó nhưng thực tế không thuộc lớp đó.

Công thức tính precision là:

$$Precision = \frac{TP}{(TP + FP)} \quad (PT\ 3.2)$$

Độ chính xác cao cho thấy tỷ lệ dự đoán đúng của mô hình trong số tất cả các dự đoán thuộc một lớp cụ thể là cao. Tuy nhiên, precision không đánh giá được tỷ lệ dự đoán đúng trong số tất cả các mẫu thực sự thuộc lớp đó. Để đánh giá khía cạnh này, ta sử dụng chỉ số recall (độ bao phủ).

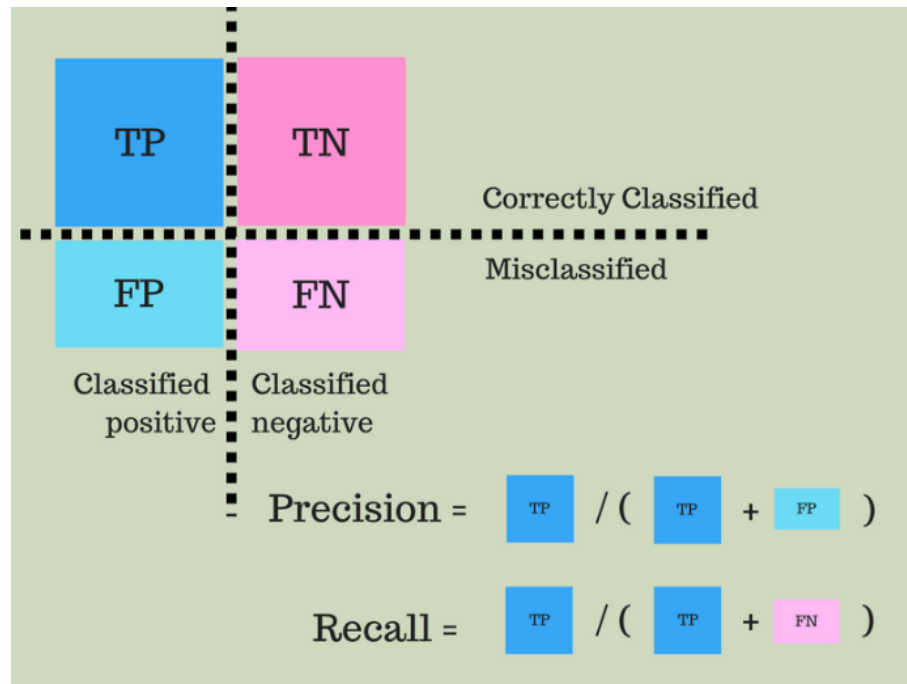
Để hiểu rõ hơn về recall, ta cần nắm vững thêm khái niệm:

- False Negative (FN): Số lượng các mẫu được dự đoán là không thuộc lớp đó nhưng thực tế lại thuộc lớp đó.

Công thức tính recall là:

$$Recall = \frac{TP}{(TP + FN)} \quad (PT\ 3.3)$$

Độ bao phủ cao cho thấy tỷ lệ dự đoán đúng của mô hình trong số tất cả các mẫu thực sự thuộc một lớp cụ thể là cao



Hình 3.3 Hình ảnh về công thức tính precision và recall

Như vậy chúng ta có hai khái niệm precision và recall với mong muốn hai giá trị này càng cao càng tốt. Tuy nhiên, trong thực tế khi chúng ta điều chỉnh mô hình để tăng recall quá mức thì có thể dẫn đến precision giảm và ngược lại. Do đó, có thêm 1 tham số F1-score để chúng ta lựa chọn. F1-score là một chỉ số tổng hợp giữa precision và recall, giúp đánh giá hiệu suất của mô hình một cách toàn diện hơn:

$$F1 = 2 \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (\text{PT 3.4})$$

Chúng ta có thể căn cứ F1-score để chọn mô hình, F1-score càng cao càng tốt là lý tưởng là bằng 1 (khi precision = recall = 1).

3.3.3 Mean Average Precision

Đường cong Precision-Recall (PR curve) biểu diễn mối quan hệ giữa Precision (độ chính xác) và Recall (độ bao phủ) khi thay đổi ngưỡng quyết định của mô hình phát hiện đối tượng.

AP (Average Precision) đo lường tỷ lệ giữa tổng Precision tại các điểm Recall khác nhau trên đường cong Precision-Recall (PR curve) và số lượng điểm Recall.

Công thức tính AP như sau:

$$AP = \sum (Precision * \Delta Recall) \quad (PT\ 3.5)$$

Trong đó, tổng được tính trên tất cả các điểm Recall và $\Delta Recall$ là sự thay đổi của Recall giữa hai điểm liên kế trên đường cong PR.

AP có giá trị trong khoảng từ 0 đến 1, với 0 đại diện cho mô hình hoàn toàn không chính xác và 1 đại diện cho mô hình hoàn hảo. Để đánh giá hiệu suất của mô hình phát hiện đối tượng trên nhiều lớp đối tượng khác nhau, người ta thường sử dụng chỉ số mAP (mean Average Precision), tức trung bình của AP trên tất cả các lớp đối tượng.

Công thức tính mAP là:

$$mAP = \frac{1}{c} \sum_{i=1}^c AP_i \quad (PT\ 3.6)$$

Trong đó c là số lớp đối tượng.

mAP cũng có giá trị trong khoảng từ 0 đến 1, với 0 đại diện cho mô hình hoàn toàn không chính xác và 1 đại diện cho mô hình hoàn hảo. Chỉ số mAP cho phép đánh giá tổng quan hiệu suất của mô hình phát hiện đối tượng trên tất cả các lớp đối tượng mà mô hình đang xử lý.

Các giá trị mAP thường được sử dụng là mAP 0.5 hay mAP 0.5:0.95.

mAP50-95 (mean Average Precision từ IoU 0.5 đến 0.95) là một chỉ số đánh giá hiệu suất của mô hình phát hiện đối tượng (object detection) dựa trên giá trị mAP (mean Average Precision) ở nhiều ngưỡng IoU (Intersection over Union) khác nhau.

Trong mAP50-95, mAP được tính ở các ngưỡng IoU từ 0.5 đến 0.95 với bước nhảy là 0.05 (tức là 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9 và 0.95). Sau đó, lấy trung bình cộng của các giá trị mAP ở các ngưỡng IoU này để tính mAP50-95. mAP50-95 là một chỉ số đánh giá phổ biến vì nó cung cấp cái nhìn tổng quát về hiệu suất của mô hình ở nhiều mức độ chính xác khác nhau. Chỉ số này giúp đánh giá mô hình không chỉ dựa trên việc phát hiện đúng đối tượng mà còn đánh giá độ chính xác của khung giới hạn dự đoán.

Công thức tính mAP50-95:

$$mAP_{50-95} = \frac{1}{10} \sum mAP_i \quad (PT\ 3.7)$$

Trong đó i chạy từ 0.5 đến 0.95 với bước nhảy là 0.05.

mAP_{50-95} càng cao thì mô hình càng tốt. Dựa trên yêu cầu chất lượng và độ chính xác mong muốn, chúng ta có thể đặt ra ngưỡng mAP_{50-95} để coi là mô hình tốt, trong bài viết này chúng ta mong muốn $mAP_{50-95} > 0.7$.

3.4 Quá trình huấn luyện các mô hình

Dữ liệu ảnh sau khi được dán nhãn, tiền xử lý và tăng cường dữ liệu sẽ được lưu về đúng định dạng phù hợp với mô hình YOLOv5

Cơ sở dữ liệu sau khi được tăng cường, được chia làm 3 phần: 6027 ảnh (90%) cho việc huấn luyện, 505 ảnh (10%) cho việc kiểm tra model huấn luyện (Validation set). Bộ dữ liệu cân bằng số lượng hình ảnh của mỗi loại tế bào bạch cầu trong cơ sở dữ liệu để cải thiện độ chính xác của mô hình và tăng tốc độ hội tụ.

Để đảm bảo công bằng trong việc đánh giá độ chính xác của các mô hình, tham số trong quá trình huấn luyện các mô hình được thiết lập như sau:

a. Batch size:

- Là số lượng mẫu dữ liệu được sử dụng để tính đạo hàm và cập nhật trọng số trong mỗi lần lặp (iteration) của quá trình huấn luyện. Việc lựa chọn batch size phù hợp trong quá trình huấn luyện mô hình có ảnh hưởng đến tốc độ hội tụ, độ chính xác và yêu cầu về bộ nhớ.
- Kích thước batch nhỏ sẽ giúp mô hình hội tụ nhanh hơn và có thể giúp mô hình thoát khỏi các điểm cực tiểu địa phương, nhưng đòi hỏi nhiều lần lặp hơn và có thể dẫn đến sự không ổn định trong quá trình huấn luyện.
- Kích thước batch lớn sẽ giúp giảm sự không ổn định, nhưng có thể làm chậm quá trình hội tụ và đòi hỏi nhiều bộ nhớ hơn.
- Giá trị batch size được lựa chọn là batch-size = 128.

b. Epochs:

- Chỉ số Epochs trong quá trình huấn luyện mô hình học máy đề cập đến số lần mô hình sẽ duyệt qua toàn bộ tập dữ liệu huấn luyện. Mỗi

epoch bao gồm việc truyền tất cả các mẫu dữ liệu qua mạng nơ-ron và cập nhật trọng số của mô hình dựa trên sai số giữa dự đoán của mô hình và giá trị thực tế.

- Số lượng epochs cần thiết để huấn luyện mô hình phụ thuộc vào bài toán cụ thể và kiến trúc mô hình. Việc huấn luyện mô hình với quá ít epochs có thể dẫn đến underfitting, khi đó mô hình không học được đủ thông tin từ dữ liệu huấn luyện và hiệu suất kém trên tập kiểm tra. Ngược lại, việc huấn luyện mô hình với quá nhiều epochs có thể dẫn đến overfitting, khi đó mô hình trở nên quá phức tạp và không khái quát hóa tốt trên dữ liệu mới.
- Việc chọn số lượng epochs phù hợp là một phần quan trọng của quá trình tinh chỉnh siêu tham số (hyperparameter tuning). Để tránh overfitting, chúng ta sử dụng kỹ thuật kiểm soát sớm (early-stopping), tức dừng quá trình huấn luyện khi chất lượng của mô hình trên tập kiểm định không cải thiện nữa sau một số lượng epochs xác định.
- Giá trị epochs được lựa chọn là epochs=2000.

c. Patience

- Patience là một siêu tham số (hyperparameter) trong early stopping, chỉ ra số lượng epoch liên tiếp mà mô hình được cho phép không cải thiện độ chính xác trên tập dữ liệu kiểm định (validation set) trước khi quá trình huấn luyện bị dừng lại.
- Giá trị epochs được lựa chọn là Patience=300

d. Img

Ảnh đầu vào. Kích thước Img = 416 x 416 pixel

e. Data.yaml

Đây là file cài đặt đường dẫn đến tập ảnh cần đưa vào huấn luyện mô hình, số lớp cần phân loại và tên các lớp đó theo đúng thứ tự các lớp được dán nhãn. Trong luận văn này chúng ta phân loại 6 lớp với tên theo thứ tự lần lượt là: basophil, diseased, eosinophil, lymphocyte, monocyte, neutrophil

f. yolov5x.yaml

Đây là tệp cấu hình mô hình cần huấn luyện, đây là mô hình lớn nhất thời gian huấn luyện lâu nhất nhưng đưa ra kết quả chính xác nhất của mô hình YOLOv5

Trong quá trình huấn luyện, vì tế bào bạch cầu bất thường rất giống tế bào lymphocyte và monocyte do đó phải tùy chỉnh huấn luyện lại nhiều lần và thay đổi chỉ số learning rate.

3.5 Đánh giá hiệu quả của từng mô hình

3.5.1 Kết quả thử nghiệm mô hình YOLOv5

Sau khi huấn luyện mô hình 472/2000 epoch, mô hình đã dừng lại vì không có thay đổi trong 300 epoch. Mô hình tốt nhất tại epoch 171, với thông số (P=0.949, R=0.915, mAP50=0.94, mAP50-95=0.786)

Kết quả phân loại tế bào bạch cầu mô hình mạng YOLOv5 tại confidence threshold mặc định, ngưỡng IoU = 0.5 như sau:



Hình 3.4 Biểu đồ heatmap dự đoán các lớp của mô hình yolov5

Theo như hình vẽ, ta có thể thấy, lớp bạch cầu basophil và eosinophil được đánh phân loại dưới trạng thái true position (TP) là khá cao 0.98 (98%). Phân loại trạng thái TP với lớp bạch cầu neutrophil và diseased lần lượt là 0.94 và 0.93 cũng là khá cao. Bạch cầu monocyte và lymphocyte là 0.84 và 0.87, 2 loại bạch cầu này

vẫn bị nhầm lẫn với tế bào bạch cầu bất thường diseased. Nguyên nhân do 2 loại bạch cầu này có đặc điểm hình dáng, màu sắc giống với tế bào bất thường.

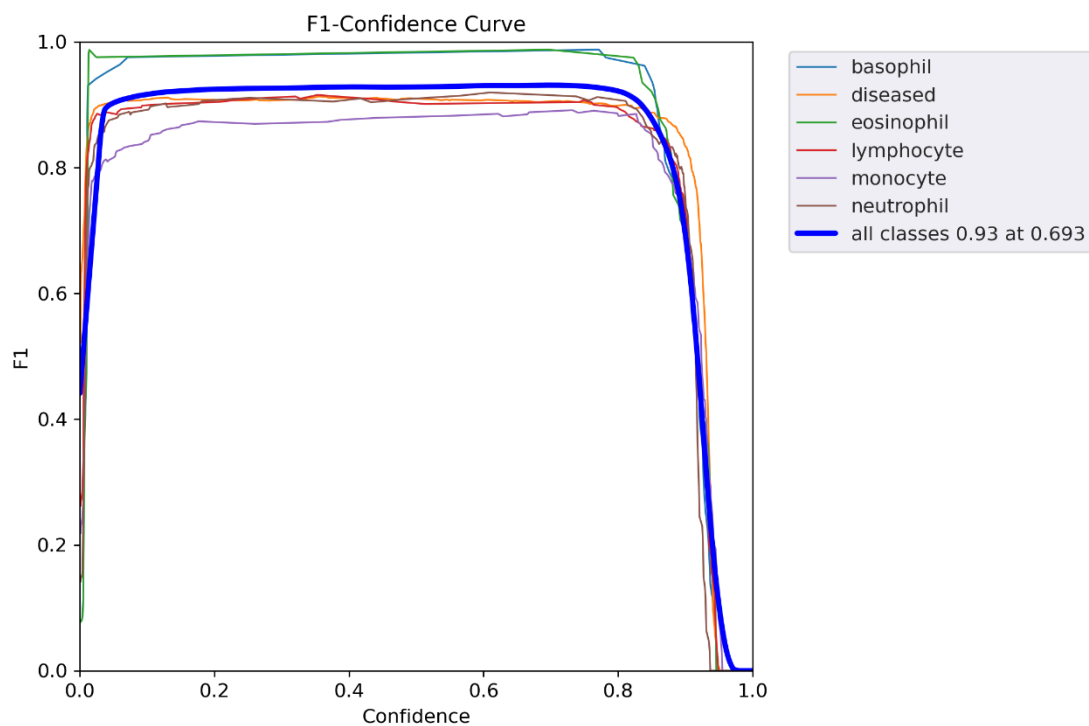
Kết quả các chỉ số đánh giá hiệu quả mô hình mạng YOLOv5 tại confidence threshold mặc định, ngưỡng IoU = 0.5 như sau:

Bảng 3.1 Đánh giá độ hiệu quả mô hình mạng YOLOv5

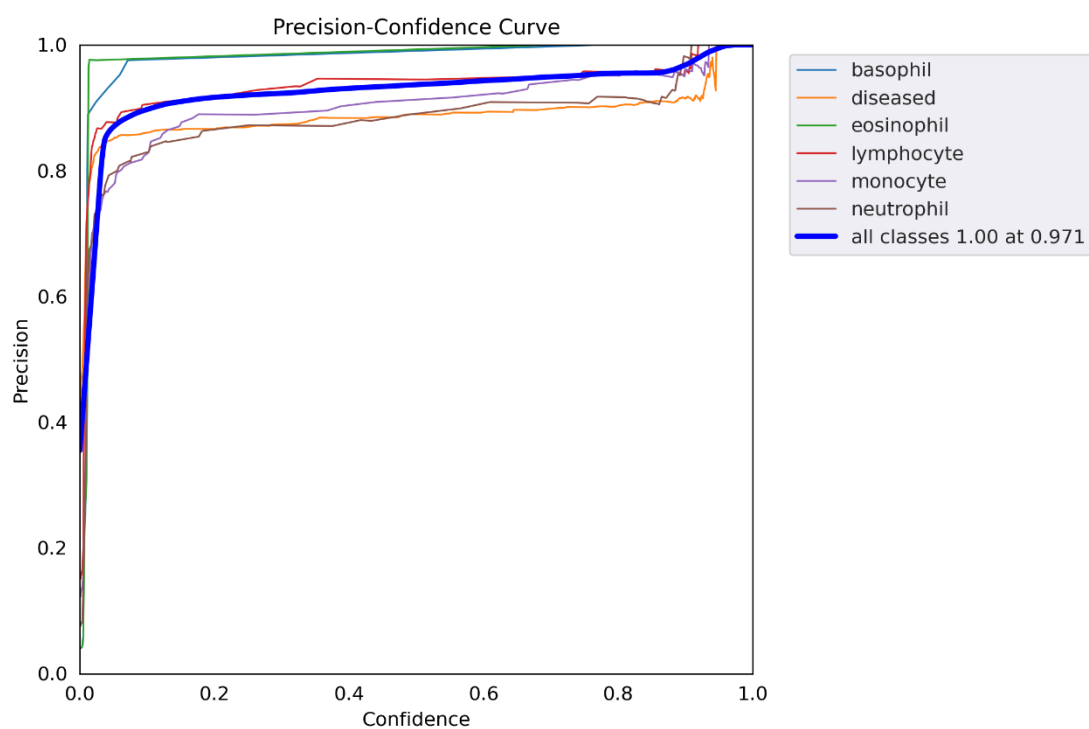
Class	Precision (P)	Recall (R)	F1-score	mAP50	mAP50-95
All	0.949	0.915	0.932	0.94	0.786
basophil	0.997	0.976	0.986	0.976	0.921
diseased	0.897	0.912	0.904	0.926	0.682
eosinophil	1	0.976	0.988	0.995	0.921
lymphocyte	0.95	0.862	0.904	0.915	0.75
monocyte	0.939	0.844	0.889	0.878	0.792
neutrophil	0.908	0.923	0.915	0.95	0.648

Giá trị Precision và Recall của mô hình đạt được ở mức cao (94,9% và 91,5%). Điều này chứng minh ở mức IoU = 0.5 mô hình YOLOv5 hoạt động hiệu quả. Giá trị của Precision và Recall khá tương đồng nhau, điều này là vô cùng hợp lý khi bộ dữ liệu kiểm tra được cân bằng số lượng giữa các phân lớp. Giá trị hàm điều hòa F1_Score đạt 93.2% nằm giữa giá trị Precision và Recall giá trị kết quả Precision và Recall tin cậy. Độ chính xác trung bình mAP50 đạt mức 94%. mAP50-95 đạt mức 78.6%.

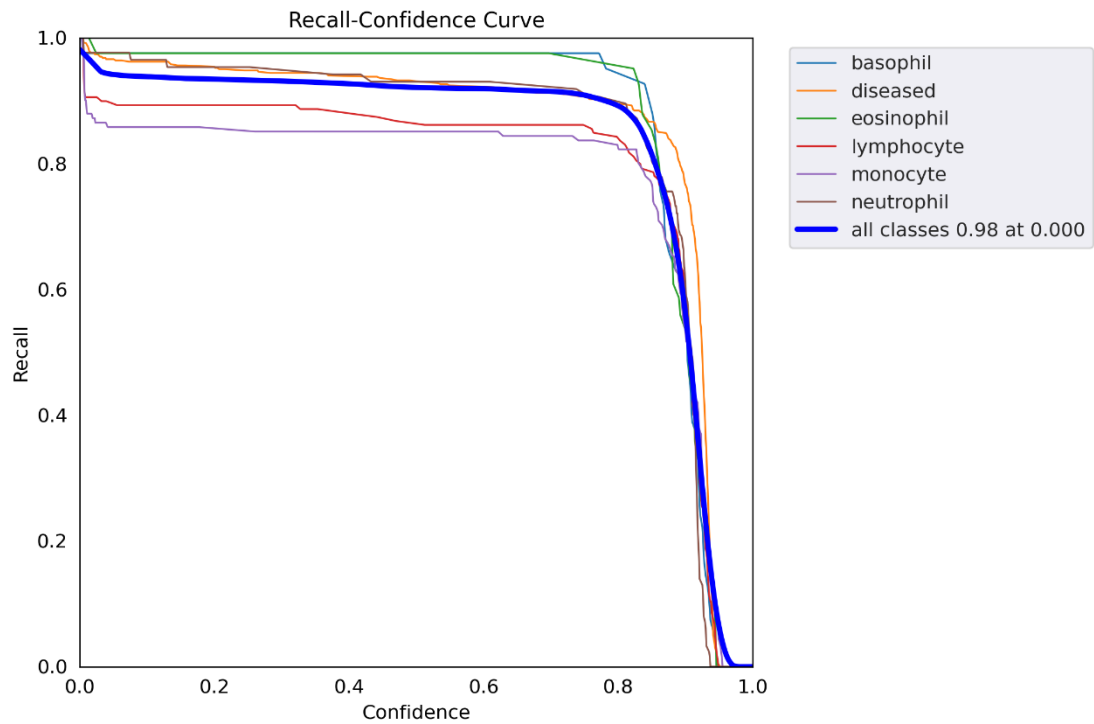
Dưới đây là hình ảnh đánh giá một số chỉ số của mô hình:



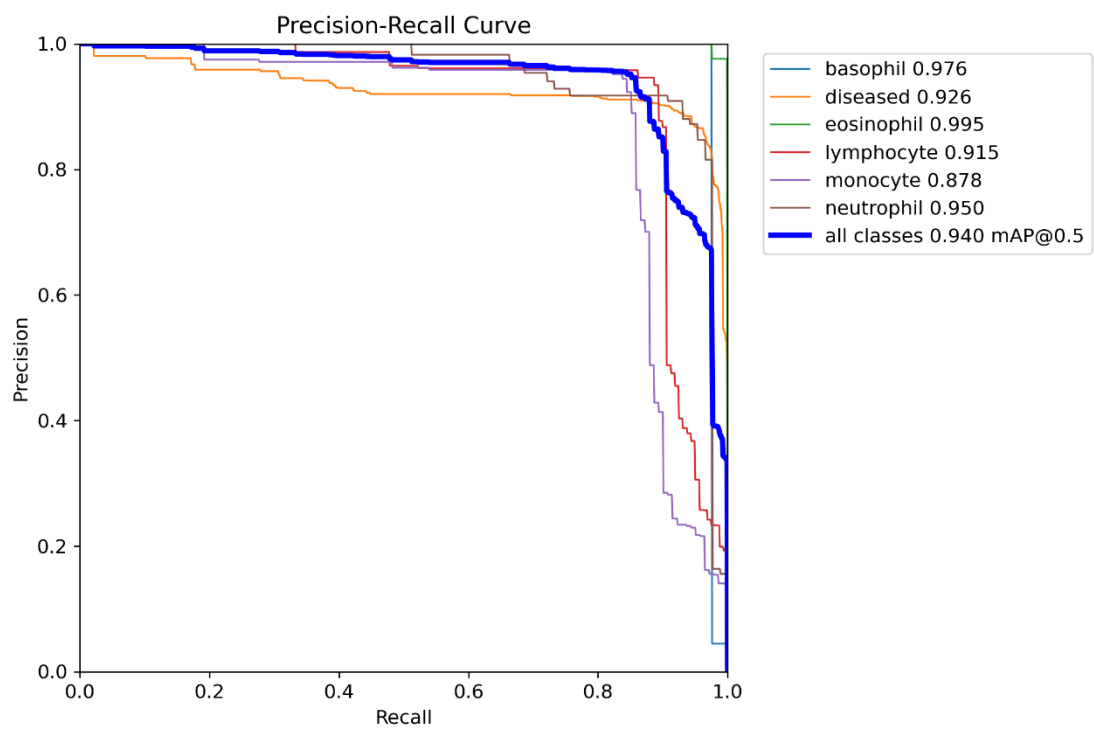
Hình 3.5 Đồ thị *F1-confidence*



Hình 3.6 Đồ thị *Precision-Confidence*

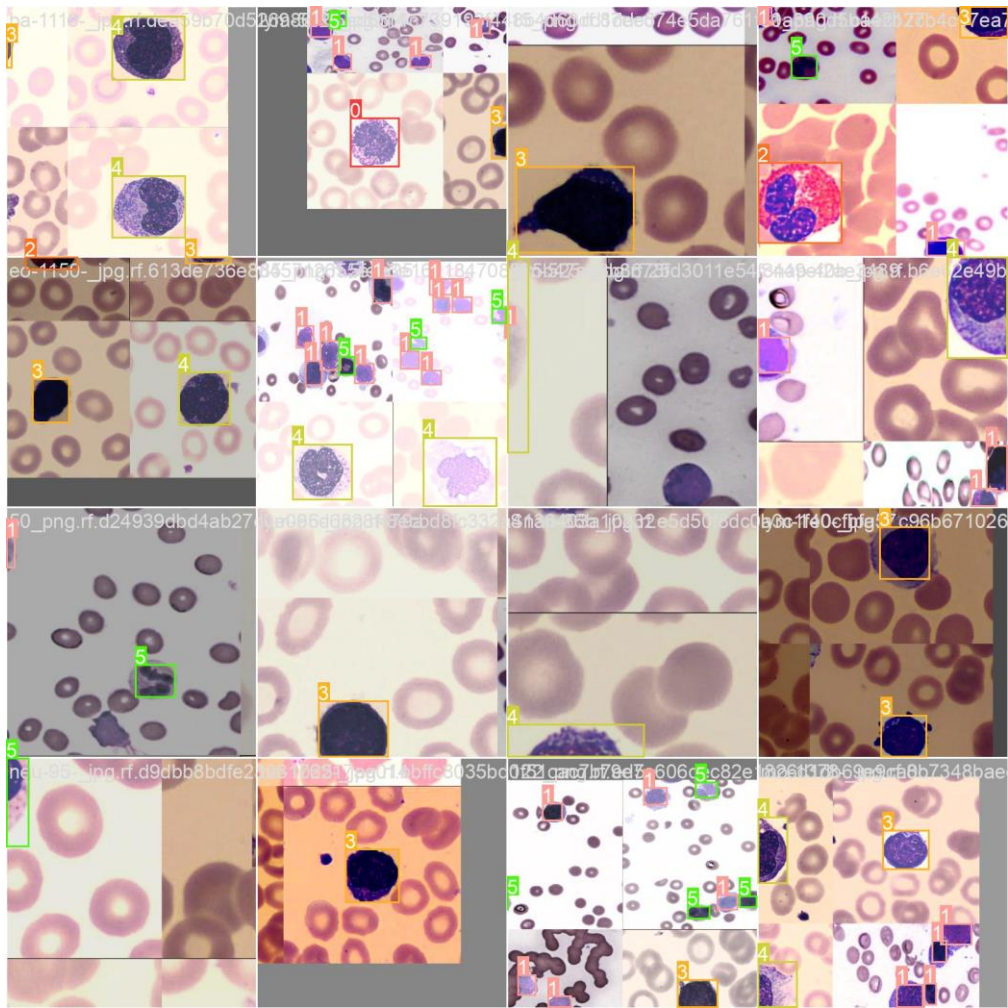


Hình 3.7 Đồ thị Recall - Confidence

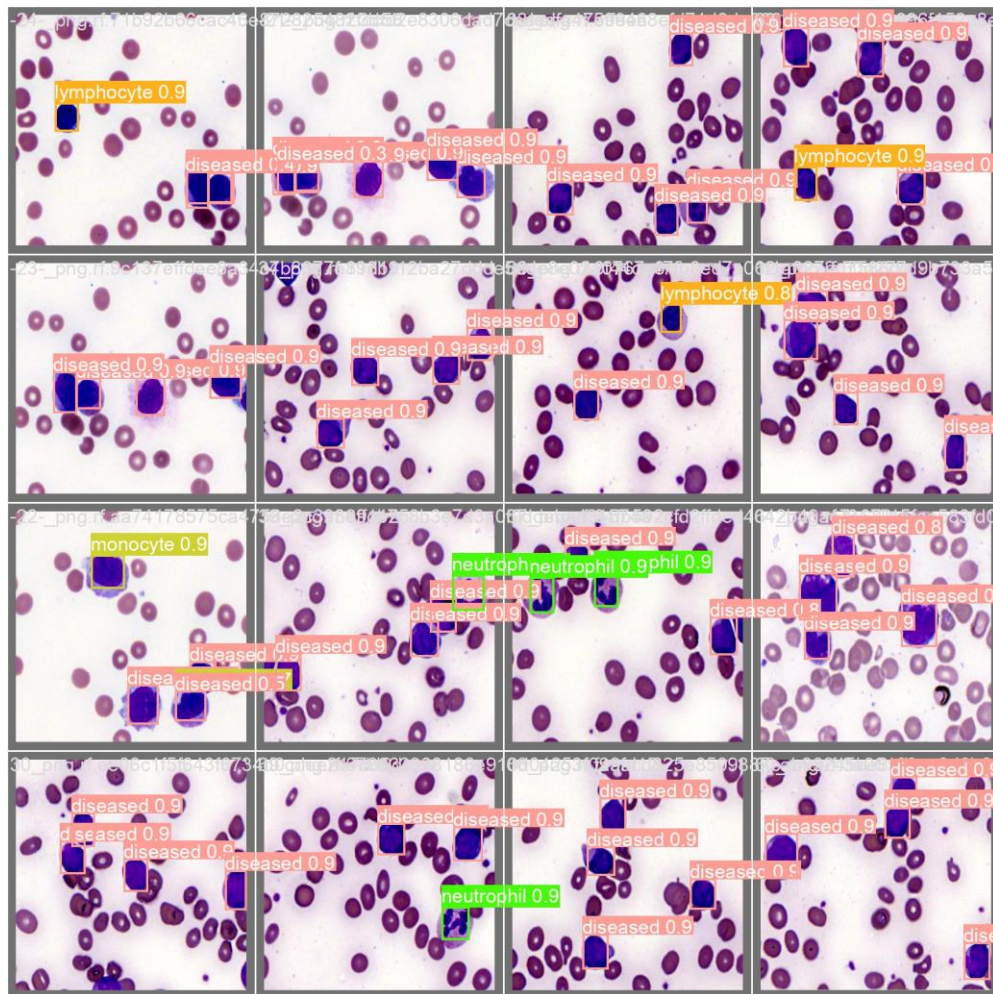


Hình 3.8 Đồ thị Precision - Recall

Một số hình ảnh train và valid mô hình:



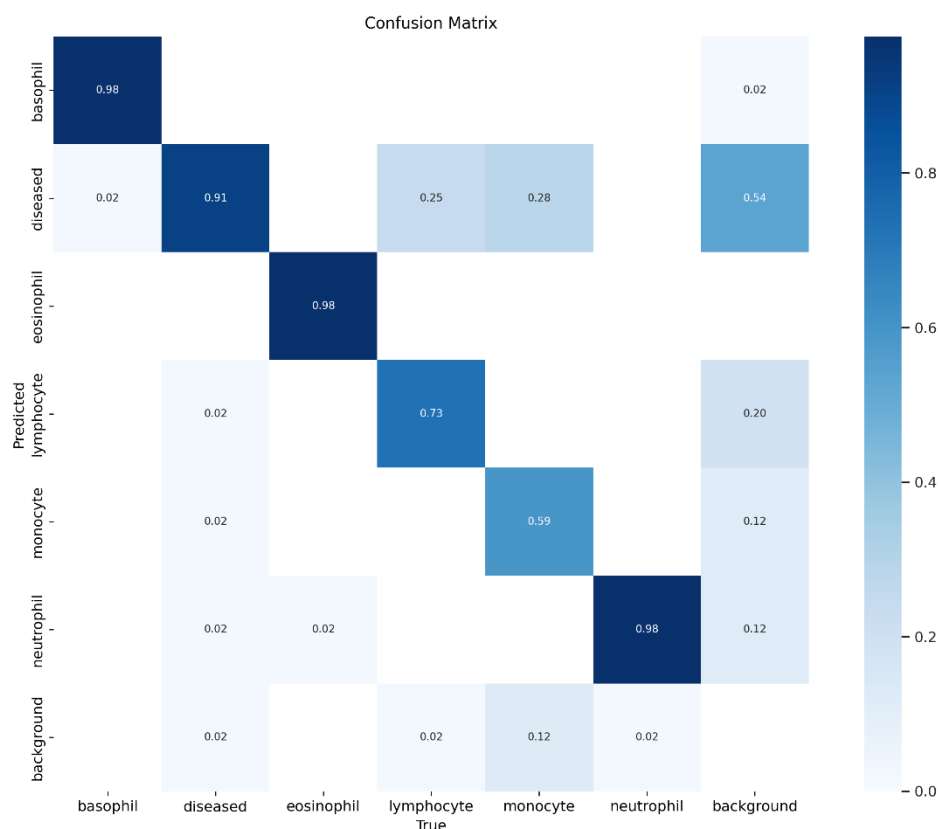
Hình 3.9 Hình ảnh huấn luyện mô hình



Hình 3.10 Hình ảnh valid mô hình

3.5.2 Kết quả thử nghiệm mô hình VGG16 kết hợp YOLOv5

Kết quả phân loại tế bào bạch cầu mô hình mạng YOLOv5 với backbone VGG16 tại confidence threshold mặc định, ngưỡng IoU = 0.5 như sau:



Hình 3.11 Đồ thị heatmap dự đoán các lớp của mô hình YOLOv5 backbone VGG16

Theo như hình vẽ, ta có thể thấy, lớp bạch cầu basophil, eosinophil và neutrophil được đánh phân loại dưới trạng thái true position (TP) là khá cao 0.98 (98%). Phân loại trạng thái TP với tế bào bạch cầu bất thường lần là 0.91. Bạch cầu monocyte và lymphocyte là 0.73 và 0.59.

Kết quả các chỉ số đánh giá hiệu quả mô hình mạng YOLOv5 với backbone VGG16 tại confidence threshold mặc định, ngưỡng IoU = 0.5 như sau:

Bảng 3.2 Đánh giá độ hiệu quả mô hình mạng YOLOv5 với backbone VGG16

Class	Precision (P)	Recall (R)	F1-score	mAP50	mAP50-95
All	0.895	0.858	0.876	0.896	0.727
basophil	0.974	0.976	0.975	0.979	0.924
diseased	0.902	0.904	0.903	0.944	0.694
eosinophil	0.991	0.976	0.983	0.994	0.925
lymphocyte	0.87	0.763	0.813	0.833	0.637
monocyte	0.768	0.57	0.654	0.669	0.528

neutrophil	0.865	0.961	0.91	0.96	0.654
------------	-------	-------	------	------	-------

Giá trị Precision và Recall của mô hình đạt được (89,5% và 85,8%). Giá trị của Precision và Recall khá tương đồng nhau, điều này là vô cùng hợp lý khi bộ dữ liệu kiểm tra được cân bằng số lượng giữa các phân lớp. Giá trị hàm điều hòa F1_Score đạt 87.6% nằm giữa giá trị Precision và Recall giá trị kết quả Precision và Recall tin cậy. Độ chính xác trung bình mAP50 đạt mức 89.6%. mAP50-95 đạt mức 72.7%.

3.6 So sánh hai mô hình

Bảng 3.3 So sánh hai mô hình

Class	Precision (P)	Recall (R)	F1- score	mAP50	mAP50- 95
YOLOv5	0.949	0.915	0.932	0.94	0.786
YOLOv5 với backbone VGG16	0.895	0.858	0.876	0.896	0.727

Qua bảng trên ta thấy mô hình YOLOv5 tốt hơn mô hình YOLOv5 với backbone VGG16.

3.7 So sánh với các mô hình khác được công khai trong các bài báo

Bảng 3.4 So sánh phương pháp đề xuất với các nghiên cứu khác

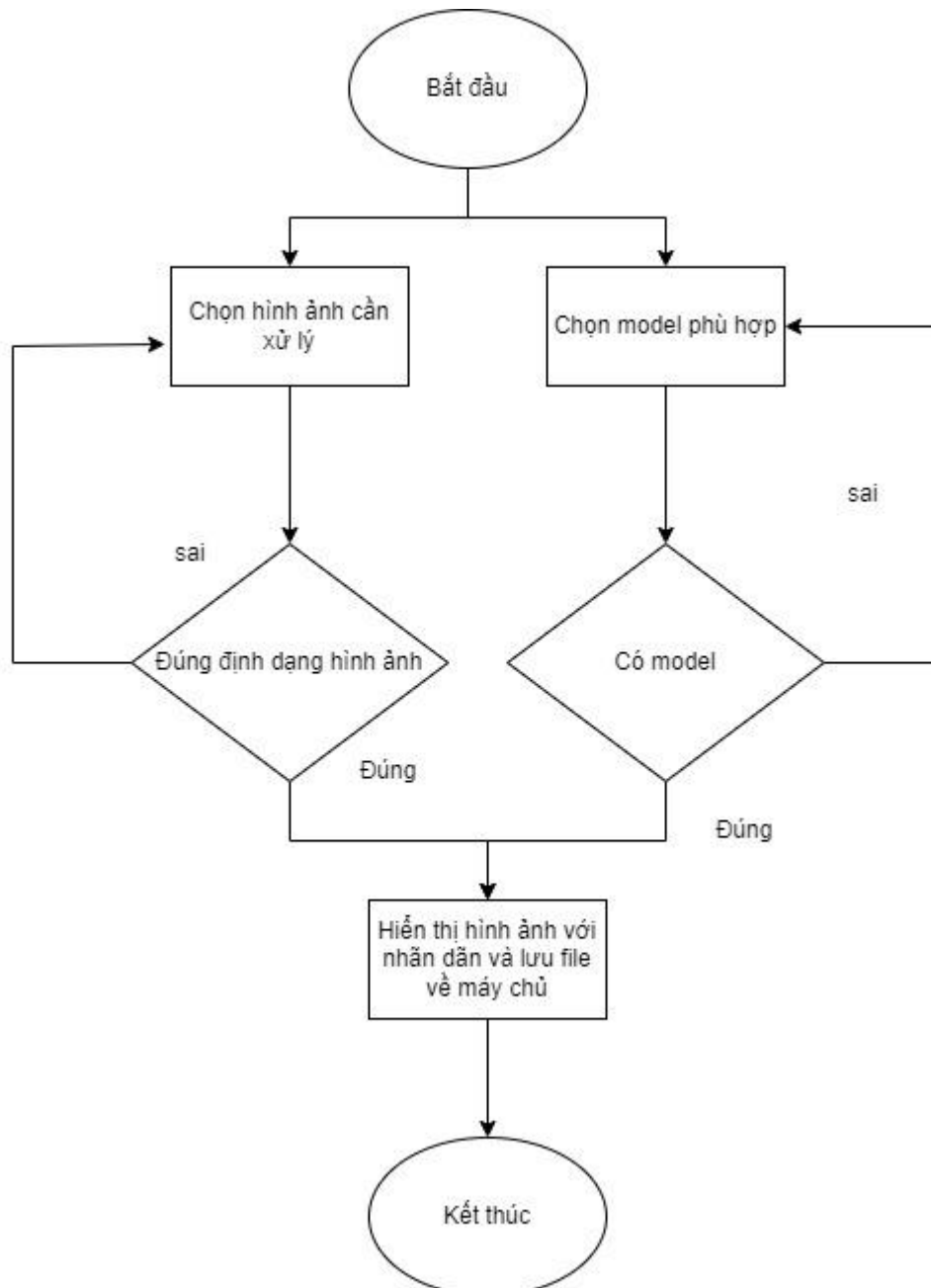
Tác giả	Mô hình	Số lớp phân loại	mAP50
Luong Duong Trong et al [29]	YOLOv4	4	97.8%
Zhao J. et al [31]	SVM, random forest	5	92.8%
Sangeetha et al. [33]	YOLOv4	3	87%
Wang et al [32]	YOLOv3	11	92%
Đề xuất	YOLOv5	6	94%

3.8 Phần mềm phân loại tế bào máu

3.8.1 Xây dựng phần mềm

Từ kết quả so sánh hai mô hình YOLOv5 và VGG16 kết hợp với YOLOv5, kết quả mô hình YOLOv5 tốt hơn, do đó phần mềm sẽ được xây dựng dựa trên mô hình YOLOv5 với tập dữ liệu train model tốt nhất của YOLOv5.

Phần mềm có backend dựa trên mô hình YOLOv5 với ngôn ngữ lập trình python và phần front end là giao diện web.



Hình 3.12 Sơ đồ giải thuật

Giao diện phần mềm:

YOLOv5 Nhận diện ảnh WBC!

Choose the image of your choice/ Chọn ảnh

Không có tệp nào được chọn

Select YOLOv5 model

#new

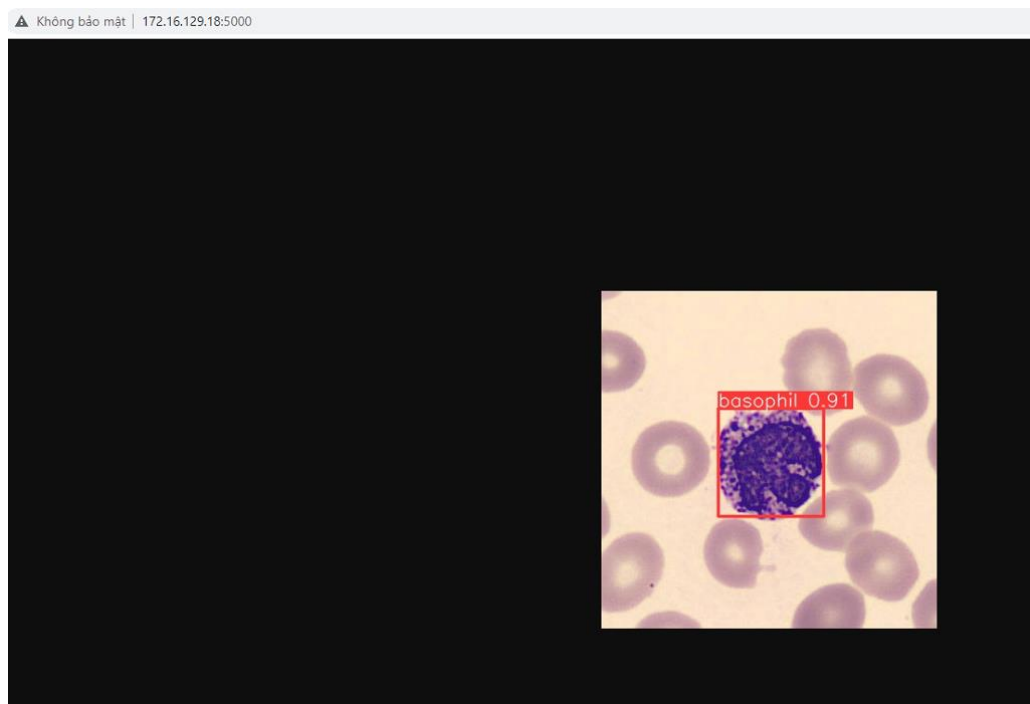
Hình 3.13 Giao diện phần mềm

3.8.2 Cách sử dụng phần mềm

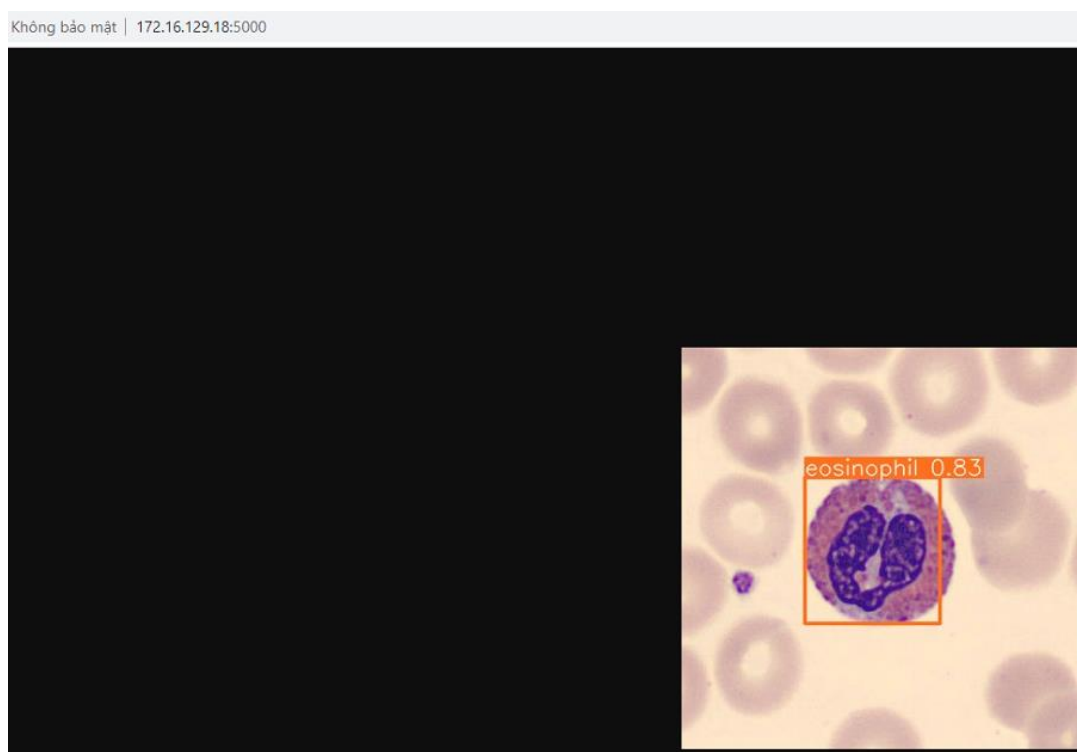
Để thực hiện việc phân loại tế bào bạch cầu, người sử dụng phần mềm cần thực hiện các bước sau:

- Bước 1: Truy cập vào địa chỉ IP trang web của phần mềm vd: 172.16.129.18:5000
- Bước 2: Nhấn vào nút “chọn tệp” và chọn đường dẫn đến file ảnh cần nhận diện.
- Bước 3: Chọn model để nhận diện
- Bước 4: Ấn submit
- Bước 5: Có thể vào thư mục share để tải ảnh đã gán nhãn về.

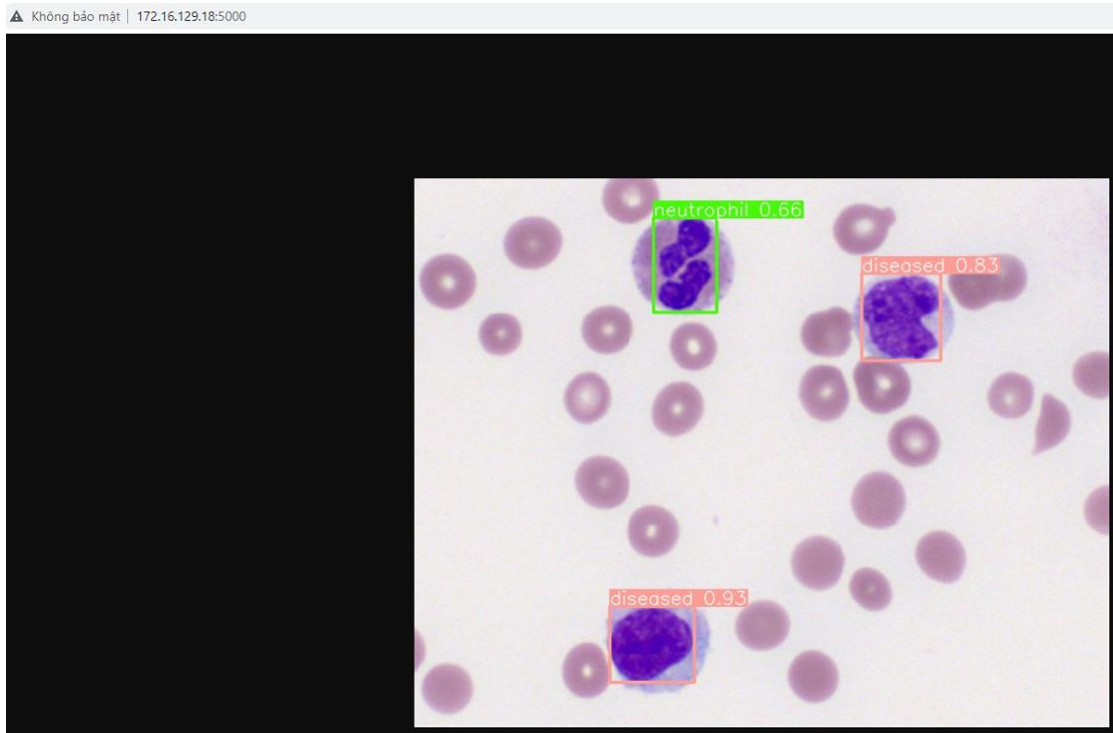
Hình ảnh của phần mềm khi nhận diện ảnh:



Hình 3.14 Kiểm thử phần mềm



Hình 3.15 Kiểm thử phần mềm



Hình 3.16 Kiểm thử phần mềm

3.9 Kết luận chương

Trong chương 3, tác giả đã trình bày quá trình xây dựng, huấn luyện và đánh giá hai mô hình phân loại tế bào máu, bao gồm mô hình YOLOv5 và mô hình kết hợp VGG16 và YOLOv5. Đối với mỗi mô hình, tác giả đã trình bày kết quả thử nghiệm và đánh giá hiệu quả thông qua các chỉ số đánh giá như Intersection over Union, Precision, Recall và Mean Average Precision.

Kết quả thử nghiệm cho thấy hai mô hình đều có khả năng phân loại tế bào máu với độ chính xác. Tuy nhiên, mô hình mô hình YOLOv5 đã có kết quả cao hơn mô hình kết hợp VGG16 và YOLOv5. Mô hình YOLOv5 đạt được thông số Precision, Recall, F1-score, mAP50 và mAP50-95 như sau: 0.949, 0.915, 0.932, 0.94, 0.786. Tác giả đã dựa vào thông số mô hình tốt nhất của mô hình YOLOv5 để xây dựng phần mềm nhận diện, phân loại tế bào máu.

KẾT LUẬN

Trong luận văn này, tác giả đã nghiên cứu và phát triển mô hình phân loại tế bào máu sử dụng các kỹ thuật học sâu, đặc biệt là mạng YOLOv5 và mô hình kết hợp VGG16 với YOLOv5. Các mô hình này được huấn luyện và kiểm tra trên cơ sở dữ liệu tế bào máu đã được thu thập và tiền xử lý từ nhiều nguồn khác nhau. Đồng thời, tác giả cũng đã xây dựng phần mềm phân loại tế bào máu dựa trên các mô hình đã nghiên cứu, giúp hỗ trợ bác sĩ và chuyên gia y tế trong việc chẩn đoán các bệnh lý máu một cách nhanh chóng và chính xác.

Cụ thể, trong chương 1, tác giả đã giới thiệu về cơ sở lý thuyết liên quan đến máu và các thành phần trong máu, bao gồm thành phần tế bào và thành phần huyết tương. Ngoài ra, tác giả cũng đã trình bày về các bệnh lý máu và mục đích, ý nghĩa của việc phân loại tế bào máu trong việc chẩn đoán và điều trị bệnh lý. Cuối cùng, tác giả đã tổng hợp một số phương pháp phân loại tế bào máu đang được sử dụng trong bệnh viện, nhằm làm rõ những hạn chế và tiềm năng của các phương pháp này.

Chương 2 của luận văn tập trung vào việc xây dựng mô hình huấn luyện phát hiện, phân loại tế bào máu. tác giả đã phân tích và trình bày mô hình mạng YOLOv5, mô hình mạng VGG16, cũng như mô hình kết hợp VGG16 và YOLOv5. Việc kết hợp hai mô hình này nhằm thử nghiệm, kiểm tra tính chính xác, với mong muốn mô hình được tốt hơn, chạy nhanh hơn trong việc phân loại tế bào máu.

Chương 3 trình bày về kết quả và bàn luận liên quan đến quá trình huấn luyện và đánh giá các mô hình đã nghiên cứu. Tác giả đã mô tả cơ sở dữ liệu huấn luyện, bao gồm quá trình thu thập, tiền xử lý và tăng cường dữ liệu, cũng như các công cụ lập trình sử dụng trong quá trình xây dựng và huấn luyện mô hình. Tác giả cũng đã trình bày các chỉ số đánh giá mô hình như Intersection over Union, Precision, Recall và Mean Average Precision, giúp đánh giá hiệu quả của từng mô hình một cách khách quan và chính xác.

Kết quả thử nghiệm cho thấy hai mô hình đều có khả năng phân loại tế bào máu với độ chính xác cao. Tuy nhiên, mô hình kết hợp VGG16 và YOLOv5 đã cho thấy hiệu quả thấp hơn so với mô hình YOLOv5 đơn thuần. Tác giả đã so sánh kết

quả của hai mô hình, nhằm đưa ra những nhận xét và đánh giá về ưu nhược điểm của từng mô hình trong việc phân loại tế bào máu.

Trong quá trình nghiên cứu, tác giả cũng đã xây dựng phần mềm phân loại tế bào máu dựa trên mô hình YOLOv5. Phần mềm này được thiết kế với giao diện thân thiện, dễ sử dụng, giúp cho các bác sĩ và chuyên gia có thể dễ dàng phân loại tế bào máu một cách nhanh chóng và chính xác. Tác giả đã trình bày cách xây dựng phần mềm, cũng như cách sử dụng phần mềm trong việc phân loại tế bào máu.

Nhìn lại toàn bộ luận văn, tác giả có thể kết luận rằng việc ứng dụng các mô hình học sâu vào việc phân loại tế bào máu là hướng đi đầy triển vọng, có thể giúp cải thiện độ chính xác và hiệu quả trong việc chẩn đoán các bệnh lý liên quan đến máu. Những nghiên cứu này sẽ mở ra những cơ hội mới cho việc áp dụng công nghệ học máy và học sâu vào lĩnh vực y khoa, đặc biệt là trong việc phân loại tế bào máu. Sự kết hợp của các mô hình học sâu đã giúp giảm thiểu những hạn chế của các phương pháp truyền thống, đồng thời tận dụng được tiềm năng của công nghệ học máy trong việc xử lý và phân tích dữ liệu y tế.

Tuy nhiên, trong quá trình nghiên cứu, tác giả cũng nhận thấy một số hạn chế và khó khăn. Việc thu thập dữ liệu tế bào máu chất lượng cao và đa dạng là một thách thức không nhỏ, đòi hỏi sự hợp tác chặt chẽ giữa các bên liên quan. Ngoài ra, các mô hình học sâu đôi khi còn gặp phải vấn đề về độ phức tạp và khả năng diễn giải kết quả, dẫn đến việc ứng dụng chúng trong thực tế còn gặp nhiều hạn chế.

Để khắc phục những hạn chế này và nâng cao hiệu quả của mô hình phân loại tế bào máu, trong tương lai, tác giả đề xuất tiếp tục nghiên cứu và cải tiến các mô hình hiện tại, đồng thời tìm kiếm và thử nghiệm các mô hình học sâu mới phù hợp hơn với đặc điểm của dữ liệu tế bào máu. Bên cạnh đó, việc mở rộng cơ sở dữ liệu và áp dụng các kỹ thuật tiền xử lý dữ liệu hiệu quả hơn cũng sẽ góp phần cải thiện độ chính xác của mô hình. Tác giả cũng đề xuất tiếp tục nghiên cứu về các phương pháp giúp tăng khả năng diễn giải của mô hình học sâu, giúp các bác sĩ và chuyên gia y tế hiểu rõ hơn về cơ chế hoạt động của mô hình và đưa ra quyết định chẩn đoán chính xác hơn.

Ngoài ra, tác giả hy vọng rằng việc kết hợp các mô hình học sâu với các phương pháp truyền thống trong việc phân loại tế bào máu sẽ tạo ra một hệ thống tổng thể mạnh mẽ và hiệu quả hơn, hỗ trợ tốt hơn cho quá trình chẩn đoán và điều trị các

bệnh lý liên quan đến máu. Việc tích hợp các mô hình học sâu vào hệ thống y tế hiện tại cũng cần phải được thực hiện một cách cẩn trọng, nhằm đảm bảo tính an toàn và bảo mật dữ liệu y tế, đồng thời tôn trọng quyền riêng tư của bệnh nhân.

Cuối cùng, tác giả tin rằng việc nâng cao chất lượng giáo dục và đào tạo về lĩnh vực học máy và học sâu trong ngành y tế sẽ giúp tạo ra một nguồn nhân lực chất lượng cao, đáp ứng nhu cầu ngày càng cao của xã hội trong việc áp dụng công nghệ vào ngành y. Đồng thời, việc tăng cường hợp tác giữa các bên liên quan, bao gồm các trung tâm nghiên cứu, các tổ chức giáo dục, các bệnh viện và các công ty công nghệ, sẽ giúp tạo ra một môi trường thuận lợi cho sự phát triển và ứng dụng các mô hình học sâu vào lĩnh vực y tế, đặc biệt là trong việc phân loại tế bào máu.

Tóm lại, thông qua việc nghiên cứu và phát triển các mô hình phân loại tế bào máu sử dụng kỹ thuật học sâu, tác giả đã đạt được một số kết quả tích cực trong việc cải thiện độ chính xác và hiệu quả của quá trình chẩn đoán các bệnh lý liên quan đến máu. Việc áp dụng công nghệ học máy và học sâu vào lĩnh vực y tế không chỉ giúp tận dụng được tiềm năng của công nghệ mà còn mang lại lợi ích thiết thực cho bệnh nhân, các bác sĩ và chuyên gia y tế. Tác giả hi vọng rằng những nỗ lực nghiên cứu trong lĩnh vực này sẽ tiếp tục phát triển và đóng góp vào sự tiến bộ của ngành y tế nói chung và việc chẩn đoán, điều trị các bệnh lý liên quan đến máu nói riêng.

TÀI LIỆU THAM KHẢO

- [1] N. A. Trí, Cẩm nang hiến máu tình nguyện, Bộ Y tế, Viện huyết, 2007.
- [2] G. N. N. Lanh, Sinh lý bệnh học, Nhà xuất bản Đại học Y Hà Nội, 2012.
- [3] "<https://niviki.com/phan-biet-ai-machine-learning-va-deep-learning/>," [Online]. [Accessed 10 03 2023].
- [4] "<https://vienhuyethoc.vn/mau-gom-nhung-thanh-phan-nao/>," [Online]. [Accessed 10 03 2023].
- [5] "<https://medium.com/analytics-vidhya/object-detection-algorithm-yolo-v5-architecture-89e0a35472ef>," [Online]. [Accessed 10 03 2023].
- [6] "<https://github.com/ultralytics/yolov5>," [Online]. [Accessed 10 03 2023].
- [7] "<https://iq.opengenus.org/yolov5/>," [Online]. [Accessed 10 03 2023].
- [8] "<https://miai.vn/2021/10/14/thu-tim-hieu-ve-map-do-luong-object-detection-model/>," [Online]. [Accessed 12 03 2023].
- [9] Ryabchykov, "Leukocyte subtypes classification by means of image," Federated Conference on Computer Science and information, Poland, 2016.
- [10] Khamael, Classification of White Blood Cells Using L-Moments, New Zealand, 2018.
- [11] Sajjad, Leukocytes classification and segmentation in microscopic, IEEE, 2016.
- [12] Yildirim, Classification of White Blood Cells by Deep Learning, 2019.
- [13] L. Y, Classification of White Blood Cells by Deep Learning, 2021.
- [14] Khan, Sajjad, Hussain, Ullah, Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images, IEEE, 2020.
- [15] Habibzadeh, Habibzadeh, Fevens, Comparative study of shape, intensity and texture features and support vector machine for white blood cell classification, Journal of Theoretical and Applied Computer Science, 2013.
- [16] Y. W, Automatic classification of leukocytes using deep neural network, China: IEEE 12th International Conference, 2017.
- [17] Macawile, White blood cell classification and counting using convolutional neural network, Japan: 3rd International Conference on Control and Robotics Engineering, 2018.
- [18] Jiang M, Cheng L, Qin FW, Du L, Zhang M, White Blood Cells Classification with Deep Convolutional Neural Networks, 2018: International Journal of Pattern Recognition and Artificial Intelligence.
- [19] F. A. Redmon J, YOLOv3: An Incremental Improvement. Preprint, 2018.

- [20] He K, Gkioxari G, Dollar P, Girshick R, Mask R-CNN, arXiv, 2018.
- [21] Zou ZX, Shi ZW Guo YH, and Ye JP, Object Detection in 20 Years: A Survey, arXiv, 2019.
- [22] Wang Q, Wang J, Zhou M, Li Q, Wen Y, Chu J, A 3D attention networks for classification of white blood cells from microscopy hyperspectral images, Opt. Laser Technol, 2021.
- [23] Wang Q, Bi S, Sun M, Wang Y, Wang D, Yang S, Deep learning approach to peripheral leukocyte recognition, PLoS ONE.
- [24] urasyeera Rohaziat, Mohd Razali Md Tomari, Wan Nurshazwani Wan Zakaria, White Blood Cells Detection using YOLOv3 with CNN Feature Extraction Models, International Journal of Advanced Computer Science and Applications, 2020.
- [25] N. Praveen, N. S. Pun, S. K. Sonbhadra, S. Agarwal, M. Syafrullah and K. Adiyarta, White blood cell subtype detection and classification, 8th International Conference on Electrical Engineering, Computer Science and Informatics, 2021.
- [26] Sangeetha, Vadivelu, Sini Anna Alex, A. Parkavi and Veena V. Pattankar, AN IMPROVED BLOOD CELL TYPE DETECTION AND CLASSIFICATION BY YOLOV4 DEEP LEARNING MODEL, 2021.
- [27] Ramya. A, Venkateswara Gupta Pola, Dr. Amrutham Bhavya Vaishnavi, Sai Suraj Karra, Comparison of YOLOv3, YOLOv4 and YOLOv5 Performance for Detection of Blood Cells, International Research Journal of Engineering and Technology, 2021.
- [28] SuNT, "https://tiensu.github.io/blog/66_object_detection_summary/," [Online]. [Accessed 15 03 2023].
- [29] Luong Duong Trong, Hung Pham Manh, Duc Nguyen Minh, Yolov4 in White Blood Cell Classification, Singapore: Springer, 2022.
- [30] Duong Trong Luong; Dao Duy Anh; Tran Xuan Thang; Hoang Thi Lan Huong; Tran Thuy Hanh; Duong Minh Khanh, Distinguish normal white blood cells from leukemia cells by detection, classification, and counting blood cells using YOLOv5, Da Nang, Vietnam: IEEE, 2022.
- [31] J. Zhao, M. Zhang, Z. Zhou, J. Chu, F. Cao, Automatic detection and classification of, Medical & biological engineering & computing, 2016.
- [32] Wang Q, Bi S, Sun M, Wang Y, Wang D, Yang S, Deep learning approach to peripheral leukocyte recognition, 2019.
- [33] Sangeetha, Vadivelu, Sini Anna Alex, A. Parkavi and Veena V. Pattankar, AN IMPROVED BLOOD CELL TYPE DETECTION AND CLASSIFICATION BY YOLOV4 DEEP LEARNING MODEL, 2021.