

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH

NGUYỄN ĐÌNH QUÂN – 22521184

NGUYỄN HÙNG PHÁT – 22521074

KHÓA LUẬN TỐT NGHIỆP
PHÁT HIỆN VÀ NHẬN DẠNG VĂN BẢN TRÊN
BIỂN HIỆU TRONG VIDEO ĐƯỜNG PHỐ VIỆT NAM

**TEXT DETECTION AND RECOGNITION ON
SIGNBOARD IN VIETNAMESE STREET-VIEW VIDEO**

CỬ NHÂN NGÀNH TRÍ TUỆ NHÂN TẠO
CỬ NHÂN NGÀNH KHOA HỌC MÁY TÍNH

TP. HỒ CHÍ MINH, 2025

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH

NGUYỄN ĐÌNH QUÂN – 22521184

NGUYỄN HÙNG PHÁT – 22521074

KHÓA LUẬN TỐT NGHIỆP
PHÁT HIỆN VÀ NHẬN DẠNG VĂN BẢN TRÊN
BIỂN HIỆU TRONG VIDEO ĐƯỜNG PHỐ VIỆT NAM

**TEXT DETECTION AND RECOGNITION ON
SIGNBOARD IN VIETNAMESE STREET-VIEW VIDEO**

CỬ NHÂN NGÀNH TRÍ TUỆ NHÂN TẠO
CỬ NHÂN NGÀNH KHOA HỌC MÁY TÍNH

GIẢNG VIÊN HƯỚNG DẪN
ThS. Đỗ Văn Tiến

TP. HỒ CHÍ MINH, 2025

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo quyết định số
ngày của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. - Chủ tịch.
2. - Thư ký.
3. - Uỷ viên.
4. - Uỷ viên.

LỜI CẢM ƠN

Trước tiên, em xin gửi lời cảm ơn chân thành và sâu sắc đến Ban Giám hiệu nhà trường và Khoa Khoa học Máy tính đã tạo điều kiện học tập và nghiên cứu thuận lợi trong suốt thời gian em theo học tại Trường Đại học Công nghệ Thông tin.

Em xin bày tỏ lòng biết ơn đặc biệt đến Thầy Đỗ Văn Tiến, đã trực tiếp giảng dạy và tận tình hướng dẫn em trong quá trình thực hiện đề tài khóa luận. Những định hướng, chỉ dẫn rõ ràng cùng sự hỗ trợ quý báu từ thầy đã là tiền đề quan trọng giúp em hoàn thành tốt công việc nghiên cứu và viết báo cáo đúng tiến độ. Em cũng xin cảm ơn thầy vì đã cung cấp tài liệu, giải đáp thắc mắc và luôn tạo môi trường học tập tích cực, hiệu quả.

Trong suốt quá trình thực hiện đề tài, em đã có cơ hội vận dụng những kiến thức nền tảng đã được học, đồng thời tích cực học hỏi, tìm tòi thêm các kiến thức mới. Đây là một trải nghiệm quý báu giúp em trưởng thành hơn trong tư duy và kỹ năng làm việc nghiên cứu.

Mặc dù đã nỗ lực hoàn thành đề tài với tinh thần nghiêm túc và cầu thị, nhưng do hạn chế về thời gian và kinh nghiệm, khóa luận không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự thông cảm, góp ý chân thành từ các thầy cô để em có thể tiếp tục hoàn thiện và phát triển trong tương lai.

Em xin chân thành cảm ơn!

TÓM TẮT KHÓA LUẬN

Phát hiện và nhận dạng văn bản trong ảnh ngoại cảnh là một bài toán quan trọng trong thị giác máy tính, hướng đến mục tiêu khai thác thông tin ngữ nghĩa phục vụ nhiều ứng dụng thực tiễn, chẳng hạn như hỗ trợ dẫn đường thông minh và phân tích loại hình kinh doanh đô thị. Trong đó, văn bản trên biển hiệu đóng vai trò đặc biệt quan trọng do thường chứa các thông tin mang tính định danh như tên địa điểm, cơ sở kinh doanh và loại hình dịch vụ. Tuy nhiên, bài toán phát hiện và nhận dạng văn bản trên biển hiệu nói chung đặt ra nhiều thách thức, xuất phát từ sự đa dạng về hình thức và bối cảnh của văn bản, cũng như đặc điểm phức tạp của biển hiệu trong môi trường thực tế. Khi mở rộng từ ảnh tĩnh sang dữ liệu video hành trình, các yếu tố như chuyển động của phương tiện, chất lượng hình ảnh hạn chế và sự biến đổi liên tục của điều kiện quan sát càng làm gia tăng độ phức tạp của bài toán. Đặc biệt, trong bối cảnh đường phố Việt Nam, những thách thức này thể hiện rõ hơn, trong khi các nghiên cứu tiếp cận theo hướng đầu-cuối (end-to-end) cho bài toán vẫn còn tương đối hạn chế.

Khóa luận này tập trung xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong ảnh hoặc video được ghi lại bởi camera hành trình. Quy trình xử lý (pipeline) đề xuất được thiết kế gồm ba giai đoạn chính, tương ứng với các tác vụ: phát hiện biển hiệu, phát hiện văn bản trong vùng biển hiệu và nhận dạng nội dung văn bản. Trên cơ sở khảo sát và thực nghiệm so sánh các phương pháp tiên tiến nay cho từng tác vụ con, khóa luận tiến hành lựa chọn và tích hợp các mô hình phù hợp nhằm đạt được sự cân bằng giữa độ chính xác và tính ổn định trong bối cảnh dữ liệu thực tế.

Các thực nghiệm được thực hiện trên tập dữ liệu SignboardText mở rộng, với đánh giá hiệu suất theo cách tiếp cận đầu-cuối (end-to-end) trên các tập con chứa văn bản

tiếng Việt và tiếng Anh. Nhằm đánh giá hiệu suất trong bối cảnh mục tiêu là đường phố Việt Nam, phân tích được tập trung trên hai tập con tiếng Việt. Kết quả thực nghiệm cho thấy quy trình xử lý (pipeline) kết hợp RTDETRv2 cho phát hiện biển hiệu, YOLOv8-OBB cho phát hiện văn bản và PARSeq cho nhận dạng văn bản đạt chỉ số $Hmean$ trong giai đoạn phát hiện văn bản lần lượt là 89.64% trên VietSignboard và 89.79% trên Vin. Đồng thời, hiệu suất nhận dạng văn bản theo cách tiếp cận đầu-cuối (end-to-end) đạt chỉ số $Hmean_{e2e}$ tương ứng là 72.32% và 72.23%. Các kết quả này cho thấy tính hiệu quả của quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất. Bên cạnh đó, việc khảo sát bước căn chỉnh biển hiệu cho thấy tiềm năng cải thiện hiệu suất trong các trường hợp biển hiệu có góc nghiêng lớn hoặc hình dạng không chuẩn.

Tóm lại, những đóng góp chính của khóa luận bao gồm xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video hành trình, hướng tới bối cảnh đường phố Việt Nam, đồng thời cung cấp các phân tích thực nghiệm có giá trị, làm cơ sở cho các nghiên cứu và ứng dụng tiếp theo trong lĩnh vực trích xuất thông tin từ ảnh hoặc video ngoại cảnh.

Mục lục

LỜI CẢM ƠN	iv
Tóm tắt khóa luận	v
Mục lục	vii
Danh mục hình	viii
Danh mục bảng	ix
Danh mục từ viết tắt	x
1 TỔNG QUAN	1
1.1 Đặt vấn đề	1
1.2 Mục tiêu và phạm vi	6
1.2.1 Mục tiêu	6
1.2.2 Phạm vi	7
1.3 Đóng góp của khóa luận	8
1.4 Cấu trúc khóa luận	9
2 CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN	10
2.1 Tổng quan và ý nghĩa thực tiễn của bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video đường phố Việt Nam	10
2.2 Phát hiện đối tượng	11
2.2.1 Cơ sở và hướng tiếp cận chung	11

2.2.2	Các nghiên cứu liên quan	11
2.3	Phát hiện và nhận dạng văn bản ngoại cảnh (Scene Text Detection and Recognition - STDR)	13
2.3.1	Phát hiện văn bản ngoại cảnh (Scene Text Detection - STD) . . .	13
2.3.1.1	Cơ sở và hướng tiếp cận chung	13
2.3.1.2	Các nghiên cứu liên quan	14
2.3.2	Nhận dạng văn bản ngoại cảnh (Scene Text Recognition - STR) .	16
2.3.2.1	Cơ sở và hướng tiếp cận chung	16
2.3.2.2	Các nghiên cứu liên quan	16
2.3.3	Nhận dạng văn bản ngoại cảnh đầu–cuối (End-to-End Scene Text Recognition)	18
2.3.3.1	Cơ sở và hướng tiếp cận chung	18
2.3.3.2	Các nghiên cứu liên quan	18
3	XÂY DỰNG QUY TRÌNH XỬ LÝ ĐẦU-CUỐI CHO PHÁT HIỆN VÀ NHẬN DẠNG VĂN BẢN TRÊN BIỂN HIỆU	21
3.1	Tổng quan quy trình xử lý đầu-cuối	21
3.2	Phát hiện biển hiệu	22
3.3	Phát hiện và nhận dạng văn bản trên biển hiệu	26
3.3.1	Phát hiện văn bản	26
3.3.2	Nhận dạng văn bản	31
3.3.3	Phát hiện và nhận dạng văn bản đầu-cuối (end-to-end)	36
4	THỰC NGHIỆM VÀ ĐÁNH GIÁ	41
4.1	Tập dữ liệu	41
4.2	Thiết lập thực nghiệm	45
4.2.1	Phát hiện biển hiệu	45
4.2.2	Phát hiện và nhận dạng văn bản trên biển hiệu	46
4.2.2.1	Hướng tiếp cận hai giai đoạn (Two-Stage)	47
4.2.2.2	Hướng tiếp cận một giai đoạn (One-Stage)	48

4.2.3	Phân chia bộ dữ liệu cho tập thực nghiệm	49
4.3	Tiền xử lý dữ liệu	50
4.4	Độ đo đánh giá	51
4.4.1	Phát hiện biến hiệu	51
4.4.2	Phát hiện văn bản	52
4.4.3	Nhận dạng văn bản	53
4.4.4	Phát hiện và nhận dạng văn bản đầu-cuối (end-to-end)	53
4.5	Kết quả thực nghiệm	53
4.5.1	Kết quả mô hình phát hiện biến hiệu đã tinh chỉnh (fine-tuning)	54
4.5.2	Kết quả mô hình tiền huấn luyện đối với bài toán phát hiện văn bản	55
4.5.3	Kết quả mô hình tiền huấn luyện đối với bài toán nhận dạng văn bản	57
4.5.4	Kết quả so sánh hướng tiếp cận hai giai đoạn (two-stage) và một giai đoạn (one-stage) trong bài toán phát hiện và nhận dạng văn bản	59
4.5.5	Kết quả mô hình phát hiện văn bản đã tinh chỉnh (fine-tuned) . .	61
4.5.6	Kết quả mô hình nhận dạng văn bản đã tinh chỉnh (fine-tuned) .	62
4.5.7	Kết quả đánh giá quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất phát hiện và nhận dạng văn bản trên biến hiệu	63
4.6	Ứng dụng minh họa quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất	68
4.6.1	Mục tiêu và thiết kế ứng dụng minh họa	68
4.6.2	Triển khai quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất trên video	69
4.6.3	Kết quả minh họa và đánh giá định tính	71
5	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	74
5.1	Kết luận	74
5.2	Hướng phát triển	75

Danh mục hình

1.1	Hình ảnh minh họa văn bản trong ảnh ngoại cảnh, trong đó văn bản trên biển hiệu là các vùng văn bản nằm bên trong vùng bao của biển hiệu. . .	2
1.2	Hình ảnh minh họa sự đa dạng về phông chữ, kích thước, chữ nghệ thuật, và đa ngôn ngữ [7]	3
1.3	Hình ảnh minh họa sự đa dạng về hình dạng, kích thước, vật liệu và vị trí của biển hiệu, dựa trên bộ dữ liệu SignboardText [7]	4
1.4	Hình ảnh minh họa đầu vào và đầu ra của bài toán. Bài toán nhận đầu vào là ảnh hoặc khung hình video và trả về danh sách các biển hiệu, trong đó mỗi biển hiệu được xác định bằng vùng chứa (bounding region) và nội dung văn bản tương ứng đã được nhận dạng.	5
2.1	Hình ảnh minh họa tổng quan quy trình phát hiện đối tượng. Ảnh đầu vào được xử lý qua mạng nơ-ron để trích xuất đặc trưng và dự đoán vị trí (hộp giới hạn) cùng phân loại các đối tượng (nhãn lớp) xuất hiện trong khung hình, tham khảo từ Analytics Vidhya [38].	12
2.2	Hình ảnh minh họa quá trình nhận dạng văn bản ngoại cảnh đầu-cuối (End-to-End Scene Text Recognition), tham khảo từ [23].	18
3.1	Tổng quan quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu.	22
3.2	Kiến trúc tổng quan của YOLO [33]	23
3.3	Kiến trúc tổng quan của DETR [4]	24
3.4	Kiến trúc tổng quan của RT-DETR, được sử dụng trong RTDETRv2 [48]	24

3.5	Kiến trúc tổng quan của SegFormer [41]	26
3.6	Kiến trúc tổng quan của Mask2Former [6]	27
3.7	Kiến trúc tổng quan của PANet [40]	28
3.8	Kiến trúc tổng quan của DBNet++ [20]	28
3.9	Kiến trúc tổng quan của TextPMs [20]	29
3.10	Kiến trúc tổng quan của FAST [5]	30
3.11	Kiến trúc tổng quan của KPN [46]	31
3.12	Kiến trúc tổng quan của ViTSTR [1]	32
3.13	Kiến trúc tổng quan của PARSeq [3]	33
3.14	Kiến trúc tổng quan của CDistNet [49]	34
3.15	Kiến trúc tổng quan của SMTR [8]	35
3.16	Kiến trúc tổng quan của SVTRv2 [10]	36
3.17	Sơ đồ kiến trúc của TESTR [47]	37
3.18	Kiến trúc tổng quan của DeepSolo [43]	38
3.19	Pipeline tổng quan của UNITS [16]	39
3.20	Kiến trúc tổng quan của DNTextSpotter [31]	40
4.1	Phân bố số lượng hình ảnh trong ba tập con của SignboardText [7]	42
4.2	Phân bố số lượng thể hiện văn bản (text instances) theo cấp độ nhãn (word-level và line-level) trong các tập con của SignboardText [7]	43
4.3	Phân bố số lượng đối tượng biển hiệu theo từng tập con của SignboardText [7]	44
4.4	Hình ảnh minh họa quá trình căn chỉnh biển hiệu (signboard alignment)	47
4.5	Hình ảnh minh họa quá trình loại bỏ dấu tiếng Việt trên dữ liệu nhãn và kết quả đầu ra của mô hình	48
4.6	So sánh trực quan kết quả phát hiện văn bản trên biển hiệu trong hai trường hợp không áp dụng và có áp dụng căn chỉnh biển hiệu (signboard alignment)	66
4.7	Sơ đồ quy trình xử lý video đầu-cuối (end-to-end) trong ứng dụng minh họa cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu.	69

4.8 Kết quả suy luận của quy trình xử lý đầu-cuối (pipeline end-to-end) trong điều kiện quan sát bất lợi, với các biến hiệu ở xa và sử dụng kiểu chữ nghệ thuật.	71
4.9 Kết quả suy luận của pipeline trong điều kiện quan sát thuận lợi, khi camera tiến gần hơn tới khu vực các biến hiệu.	72
4.10 Kết quả suy luận của pipeline trong trường hợp biến hiệu có sự xuất hiện của các thành phần phi văn bản (biểu tượng, logo) xen kẽ với nội dung văn bản.	73

Danh mục bảng

4.1	Thống kê số lượng và tỷ lệ văn bản nằm trong vùng biển hiệu so với toàn bộ văn bản trong ảnh trên các tập con của SignboardText	45
4.2	Hiệu suất phát hiện biển hiệu của các mô hình với đầu ra dạng hình chữ nhật (rectangle bounding box). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	54
4.3	Hiệu suất phát hiện biển hiệu của các mô hình với đầu ra dạng vùng bao định hướng (Oriented Bounding Box - OBB). Chỉ số tốt nhất được đánh dấu đậm	55
4.4	Hiệu suất phát hiện vùng biển hiệu của các mô hình phân đoạn ngữ nghĩa. Chỉ số tốt nhất được đánh dấu đậm.	55
4.5	Hiệu suất mô hình tiền huấn luyện trong bài toán phát hiện văn bản ở cấp độ từ (word-level) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText mở rộng. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	57
4.6	Hiệu suất mô hình tiền huấn luyện trong bài toán phát hiện văn bản ở cấp độ dòng (line-level) trên tập con VietSignboard của tập dữ liệu SignboardText mở rộng. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	57

4.7 Hiệu suất các mô hình tiền huấn luyện trong bài toán nhận dạng văn bản trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText đã được mở rộng, với hai cấp độ đánh giá từ (word-level) và dòng (line-level). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	59
4.8 So sánh hiệu suất mô hình tiền huấn luyện theo hai hướng tiếp cận hai giai đoạn (two-stage) và một giai đoạn (one-stage) trong bài toán phát hiện và nhận dạng văn bản trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText, với hai cấp độ đánh giá từ (word-level) và dòng (line-level). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới	60
4.9 Hiệu suất mô hình phát hiện văn bản đã tinh chỉnh. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	62
4.10 Hiệu suất mô hình nhận dạng văn bản đã tinh chỉnh. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	63
4.11 Hiệu suất phát hiện văn bản của quy trình xử lý đầu-cuối (pipeline end-to-end) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	67
4.12 Hiệu suất nhận dạng văn bản của quy trình xử lý đầu-cuối (pipeline end-to-end) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.	67

Danh mục từ viết tắt

Từ viết tắt	Giải thích
STDR	Scene Text Detection and Recognition - Phát hiện và nhận dạng văn bản trong ảnh ngoại cảnh
STD	Scene Text Detection - Phát hiện văn bản trong ảnh ngoại cảnh
STR	Scene Text Recognition - Nhận dạng văn bản trong ảnh ngoại cảnh
OCR	Optical Character Recognition - Nhận dạng ký tự quang học
CNN	Convolutional Neural Network - Mạng nơ-ron tích chập
RNN	Recurrent Neural Network - Mạng nơ-ron hồi quy
LSTM	Long Short-Term Memory - Bộ nhớ dài-ngắn hạn
YOLO	You Only Look Once - Mô hình phát hiện đối tượng
DETR	Detection Transformer - Transformer cho phát hiện đối tượng
RT-DETR	Real-Time Detection Transformer - DETR thời gian thực
R-CNN	Region-based Convolutional Neural Network
SSD	Single Shot Detector
SegFormer	Semantic Segmentation Transformer
Mask2Former	Masked-attention Mask Transformer
DBNet	Differentiable Binarization Network
PANet	Path Aggregation Network
FAST	Faster Arbitrary-shaped Text Detector
KPN	Kernel Proposal Network
TextPMs	Text Pyramid Matcher
ViTSTR	Vision Transformer for Scene Text Recognition
PARSeq	Permuted Autoregressive Sequence Models
CDistNet	Character Distance Network
SMTR	Spatial-aware Multi-aspect Text Recognizer
SVTrv2	Single Visual model for Scene Text Recognition v2
TESTR	Text Spotting Transformer
DeepSolo	Deep Single Point Oriented Text Locator
UNITS	Unified Text Spotter
mAP	mean Average Precision - Độ chính xác trung bình
IoU	Intersection over Union - Tỷ lệ giao trên hợp
FPS	Frames Per Second - Số khung hình trên giây
F1	F1-score - Điểm F1
NMS	Non-Maximum Suppression - Triệt tiêu không cực đại
FPN	Feature Pyramid Network - Mạng kim tự tháp đặc trưng
ROI	Region of Interest - Vùng quan tâm
GPU	Graphics Processing Unit - Bộ xử lý đồ họa
API	Application Programming Interface - Giao diện lập trình ứng dụng

Chương 1

TỔNG QUAN

1.1 Đặt vấn đề

Phát hiện và nhận dạng văn bản trong ảnh ngoại cảnh (Scene Text Detection and Recognition – STDR) là một bài toán quan trọng trong thị giác máy tính, thu hút nhiều sự quan tâm nhờ tính ứng dụng rộng rãi như dịch tự động, hỗ trợ dẫn đường, hay phân tích biển báo giao thông. Với đầu vào là ảnh tĩnh hoặc các khung hình video, bài toán STDR hướng tới việc xác định vị trí xuất hiện và nội dung của văn bản (Hình 1.1).

Trong số các loại văn bản ngoại cảnh, **văn bản trên biển hiệu** (Hình 1.1) có ý nghĩa đặc biệt do thường chứa các thông tin quan trọng như tên địa điểm, cơ sở kinh doanh hoặc loại hình dịch vụ. Chính vì vậy, bài toán **phát hiện và nhận dạng văn bản trên biển hiệu** (Text Detection and Recognition on Signboard) trở thành một nhánh nghiên cứu quan trọng của STDR, với nhiều tiềm năng ứng dụng trong hệ thống dẫn đường thông minh, phân tích thông tin đô thị, và bổ sung thông tin ngữ nghĩa cho bản đồ số.

Tuy nhiên, bài toán phát hiện và nhận dạng văn bản trên biển hiệu đặt ra nhiều thách thức. Thách thức đầu tiên xuất phát từ đặc điểm của văn bản, như sự đa dạng về phông chữ, kích thước, hướng, bô cục; văn bản có thể bị nghiêng, cong, chồng chép hoặc hòa lẫn vào nền phức tạp, cùng với các phong cách thiết kế nghệ thuật và yếu tố đa ngôn ngữ (Hình 1.2). Đặc biệt đối với tiếng Việt, khó khăn còn gia tăng do hệ thống dấu thanh (sắc, huyền, hỏi, ngã, nặng) và các ký tự đặc biệt (ô, ê, ă, â, ơ, ư), làm tăng đáng kể tập ký tự cần nhận dạng và dễ gây nhầm lẫn giữa các chữ có hình dáng tương tự (ví dụ giữa



Hình 1.1: Hình ảnh minh họa văn bản trong ảnh ngoại cảnh, trong đó văn bản trên biển hiệu là các vùng văn bản nằm bên trong vùng bao của biển hiệu.

a, â, ă, ā̂.

Thách thức thứ hai bắt nguồn từ đặc điểm của biển hiệu và bối cảnh môi trường xung quanh, biển hiệu đa dạng về hình dạng, kích thước, vật liệu và thường xuất hiện ở các vị trí phức tạp trong ảnh (Hình 1.3), chẳng hạn như bị che khuất một phần, chịu ảnh hưởng của phản xạ ánh sáng, hoặc nằm trong các bối cảnh đông đúc. Theo khảo sát các nghiên cứu hiện có, cho đến nay mới chỉ có một nghiên cứu [32] tập trung vào phát hiện biển hiệu trên đường phố Việt Nam, trong khi hướng tiếp cận kết hợp cả phát hiện đối tượng biển hiệu lẫn nhận dạng nội dung văn bản trên đó vẫn còn rất ít được khai thác.

Hơn nữa, khi mở rộng phạm vi từ ảnh tĩnh sang **video hành trình**, bài toán còn phải đổi mới với những thách thức đặc thù như hiện tượng mờ do chuyển động, chất lượng hình ảnh bị giới hạn bởi camera hành trình, cùng với sự biến đổi liên tục về điều kiện



Hình 1.2: Hình ảnh minh họa sự đa dạng về phông chữ, kích thước, chữ nghệ thuật, và đa ngôn ngữ [7]

ánh sáng và góc quay. Những yếu tố này khiến nhiệm vụ phát hiện và nhận dạng văn bản trong video trở nên phức tạp hơn nhiều so với trên ảnh đơn lẻ.

Từ những thách thức nêu trên, bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong bối cảnh **video hành trình** có thể được định nghĩa một cách cụ thể như sau (hình ảnh minh họa trực quan tại Hình 1.4):

- **Đầu vào (Input):** Hình ảnh hoặc khung hình thực tế được trích xuất từ video camera hành trình trên đường phố Việt Nam, chứa các cảnh có biển hiệu trong nhiều điều kiện khác nhau, bao gồm ban ngày/ban đêm, trời nắng/mưa và các góc nhìn đa dạng.
- **Đầu ra (Output):** Với hình ảnh (hoặc khung hình video) đầu vào, hệ thống cần trả về hai thông tin chính:
 - **Vị trí của biển hiệu:** Danh sách các vùng (bounding regions) xác định vùng chứa biển hiệu trong ảnh.



Hình 1.3: Hình ảnh minh họa sự đa dạng về hình dạng, kích thước, vật liệu và vị trí của biển hiệu, dựa trên bộ dữ liệu SignboardText [7]

- Thông tin văn bản trên từng biển hiệu:** Ứng với mỗi biển hiệu, cung cấp vị trí và nội dung văn bản đã được nhận dạng trên biển hiệu đó.

(*Kết quả đầu ra có thể được trực quan hóa trực tiếp trên ảnh đầu vào hoặc tích hợp để xử lý liên tục cho luồng video.*)

Trước những thách thức thực tế và dựa trên các kết quả nghiên cứu trước đây cho thấy rằng hướng nghiên cứu kết hợp (phát hiện biển hiệu và nhận dạng văn bản) vẫn còn ít được khai thác, khóa luận này đặt ra mục tiêu xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video được quay bởi camera hành trình trên đường phố. Quy trình xử lý (pipeline)



Hình 1.4: Hình ảnh minh họa đầu vào và đầu ra của bài toán. Bài toán nhận đầu vào là ảnh hoặc khung hình video và trả về danh sách các biển hiệu, trong đó mỗi biển hiệu được xác định bằng vùng chứa (bounding region) và nội dung văn bản tương ứng đã được nhận dạng.

hướng tới việc:

- Xác định vùng chứa biển hiệu (signboard detection) và vùng chứa văn bản bên trong mỗi biển hiệu (text detection) trong từng khung hình video.
- Trích xuất và chuyển đổi nội dung văn bản từ các vùng văn bản đã phát hiện thành dạng văn bản có thể đọc được, hỗ trợ hai ngôn ngữ chính là tiếng Việt và tiếng Anh, hướng tới việc cung cấp thông tin đầu ra có ích cho các tác vụ truy xuất hoặc khai thác thông tin trong tương lai.

Để đạt được các mục tiêu trên, khóa luận sẽ tiến hành khảo sát, thực nghiệm so sánh và lựa chọn các phương pháp tiên tiến hiện nay cho từng tác vụ con, đồng thời so sánh hai hướng tiếp cận chính cho bài toán phát hiện và nhận dạng văn bản đầu-cuối (end-to-end text spotting). Các phương pháp cụ thể được xem xét bao gồm:

- **Phát hiện biển hiệu (Signboard Detection):** YOLOv8 [37], YOLOv11 [37] , DETR [4], RTDETR v2 [26], YOLOv8-OBB [37], YOLOv11-OBB [37], SegFormer [41], Mask2Former [6].
- **Phát hiện văn bản (Text Detection):** PANet [40], DBNet++ [20], TextPMs [44], FAST [5], KPN [46], YOLOv8-OBB [37], YOLOv11-OBB [37].
- **Nhận dạng văn bản (Text Recognition):** ViTSTR [1], PARSeq [3], CDistNet [49], SMTR [8], SVTRv2 [10]
- **Phát hiện và nhận dạng văn bản đầu-cuối (End-to-End Text Spotting):** TESTR [47], DeepSolo [43], UNITS [16], DNTextSpotter [31]

Trên cơ sở kết quả đánh giá và so sánh từ thực nghiệm cho từng tác vụ con, một quy trình xử lý đầu-cuối (pipeline end-to-end) sẽ được xây dựng bằng cách lựa chọn phương pháp tối ưu cho mỗi tác vụ và xác định kiến trúc hiệu quả nhất cho giai đoạn xử lý văn bản thông qua so sánh hướng tiếp cận hai giai đoạn (two-stage) (tích hợp các phương pháp phát hiện và nhận dạng văn bản đã chọn) với các mô hình đầu-cuối (end-to-end) tiên tiến.

1.2 Mục tiêu và phạm vi

1.2.1 Mục tiêu

Trong khóa luận này, sinh viên đề ra các mục tiêu như sau:

- Mở rộng và chuẩn bị tập dữ liệu ảnh tĩnh SignboardText [7] bằng cách bổ sung nhãn đối tượng biển hiệu (*signboard*), nhằm hỗ trợ đánh giá bài toán phát hiện biển hiệu. Đồng thời, thu thập dữ liệu video đường phố giúp đánh giá khả năng tổng quát hóa của quy trình xử lý (pipeline) phát hiện và nhận dạng văn bản trên biển hiệu.

- Thực nghiệm, so sánh và đánh giá một số phương pháp tiên tiến hiện nay cho từng tác vụ con (phát hiện biển hiệu, phát hiện văn bản, nhận dạng văn bản) trên tập dữ liệu được chuẩn bị, từ đó rút ra ưu điểm, nhược điểm của từng phương pháp.
- Xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video hành trình tại Việt Nam.
- Xây dựng ứng dụng minh họa nhằm trực quan hóa kết quả suy luận của quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất trên dữ liệu video hành trình tại Việt Nam.

1.2.2 Phạm vi

Phạm vi của khóa luận được giới hạn nhằm đảm bảo tính tập trung và khả thi, bao gồm các công việc sau:

- Mở rộng tập dữ liệu tập trung vào việc bổ sung nhãn đối tượng biển hiệu (signboard annotation) cho tập dữ liệu ảnh tĩnh SignboardText hiện có. Dữ liệu video được thu thập chỉ nhằm mục đích minh họa và kiểm tra tính tổng quát của mô hình, với điều kiện chính là ban ngày. Các tình huống phức tạp (ban đêm, thời tiết xấu) không nằm trong phạm vi xem xét.
- Khảo sát và thực nghiệm được giới hạn trong một tập hợp các phương pháp tiên tiến cho các hướng tiếp cận phổ biến và hiệu quả hiện nay. Việc so sánh không bao quát toàn bộ các phương pháp trong lĩnh vực, mà tập trung vào những phương pháp phù hợp và khả thi với dữ liệu và mục tiêu của khóa luận. Các phương pháp cụ thể được xem xét bao gồm:
 - **Phát hiện biển hiệu (Signboard Detection):** YOLOv8 [37], YOLOv11 [37], DETR [4], RTDETR v2 [26], YOLOv8-OBB [37], YOLOv11-OBB [37], SegFormer [41], Mask2Former [6].
 - **Phát hiện văn bản (Text Detection):** PANet [40], DBNet++ [20], TextPMs [44], FAST [5], KPN [46], YOLOv8-OBB [37], YOLOv11-OBB [37].

- **Nhận dạng văn bản (Text Recognition):** ViTSTR [1], PARSeq [3], CDistNet [49], SMTR [8], SVTRv2 [10]
- **Phát hiện và nhận dạng văn bản đầu-cuối (End-to-End Text Spotting):** TESTR [47], DeepSolo [43], UNITS [16], DNTextSpotter [31]
- Quy trình xử lý đầu-cuối (pipeline end-to-end) tập trung vào bài toán phát hiện và nhận dạng văn bản trên biển hiệu và hướng tới việc cung cấp thông tin đầu ra vị trí và nội dung văn bản, làm cơ sở cho các tác vụ truy xuất thông tin trong tương lai.
- Ứng dụng minh họa không hướng đến việc xây dựng một hệ thống hoàn chỉnh với giao diện người dùng hoặc các chức năng tương tác phức tạp. Thay vào đó, ứng dụng tập trung vào việc biểu diễn kết quả xử lý của quy trình xử lý đầu-cuối (pipeline end-to-end) dưới dạng video đầu ra đã được gán nhãn, nhằm phục vụ mục đích minh họa và đánh giá định tính trong phạm vi khóa luận.

1.3 Đóng góp của khóa luận

Các đóng góp chính của khóa luận bao gồm:

- **Mở rộng bộ dữ liệu:** Bổ sung nhãn đối tượng biển hiệu (signboard bounding box) cho tập dữ liệu ảnh tĩnh SignboardText [7], hỗ trợ thực nghiệm và đánh giá cho bài toán phát hiện biển hiệu. Đồng thời, thu thập một tập dữ liệu video hành trình thực tế để phục vụ minh họa và kiểm tra tính tổng quát của pipeline.
- **Thực nghiệm và đánh giá:** Tiến hành cài đặt, thực nghiệm và so sánh một số phương pháp tiên tiến cho ba tác vụ thành phần: phát hiện biển hiệu, phát hiện văn bản và nhận dạng văn bản. Các phương pháp cụ thể được xem xét bao gồm:
 - **Phát hiện biển hiệu (Signboard Detection):** YOLOv8 [37], YOLOv11 [37], DETR [4], RTDETR v2 [26], YOLOv8-OBB [37], YOLOv11-OBB [37], SegFormer [41], Mask2Former [6].
 - **Phát hiện văn bản (Text Detection):** PANet [40], DBNet++ [20], TextPMs [44], FAST [5], KPN [46], YOLOv8-OBB [37], YOLOv11-OBB [37].

- **Nhận dạng văn bản (Text Recognition):** ViTSTR [1], PARSeq [3], CDistNet [49], SMTR [8], SVTRv2 [10]
- **Phát hiện và nhận dạng văn bản đầu-cuối (End-to-End Text Spotting):** TESTR [47], DeepSolo [43], UNITS [16], DNTextSpotter [31]

Kết quả đánh giá đi kèm phân tích ưu và nhược điểm của các phương pháp trong bối cảnh dữ liệu tiếng Việt và cảnh quan đường phố.

- **Xây dựng quy trình xử lý đầu-cuối (pipeline end-to-end):** Trên cơ sở kết quả thực nghiệm, xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video hành trình trên đường phố Việt Nam. Quy trình xử lý (pipeline) hướng tới việc cung cấp đầu ra vị trí và nội dung văn bản, làm cơ sở cho các tác vụ truy xuất thông tin trong tương lai.

1.4 Cấu trúc khóa luận

Nội dung khóa luận được tổ chức như sau:

Chương 1: Tổng quan bài toán, mục tiêu, phạm vi và đóng góp.

Chương 2: Cơ sở lý thuyết và các nghiên cứu liên quan.

Chương 3: Thiết kế quy trình xử lý đầu-cuối (pipeline end-to-end)

Chương 4: Thực nghiệm và đánh giá.

Chương 5: Ứng dụng minh họa, kết luận và hướng phát triển.

Chương 2

CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN

2.1 Tổng quan và ý nghĩa thực tiễn của bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video đường phố Việt Nam

Trong môi trường giao thông đô thị tại Việt Nam, biển hiệu chứa đựng lượng lớn thông tin ngữ nghĩa cấp cao, phản ánh trực tiếp danh tính (tên cửa hàng, địa điểm) và loại hình kinh doanh/dịch vụ. Việc tự động trích xuất và hiểu các thông tin này từ luồng video không chỉ giúp giảm bớt việc gán nhãn dữ liệu thủ công mà còn mở ra nhiều ứng dụng thực tiễn hữu ích, có thể kể đến như:

- **Hệ thống dẫn đường thông minh:** Bổ sung thông tin các địa điểm thực tế (tên cửa hàng, địa điểm) từ biển hiệu vào hệ thống dẫn đường, giúp cải thiện độ chính xác của định vị và điều hướng.
- **Phân tích thông tin đô thị:** Tự động thống kê và phân loại các loại hình kinh doanh theo tuyến đường hoặc khu vực, phục vụ quy hoạch và nghiên cứu thị trường.

Trong bối cảnh này, để hiện thực hóa các ứng dụng trên, bài toán đặt ra nhiều thách thức kỹ thuật. Ngoài những khó khăn chung của nhận dạng văn bản trong cảnh (như đa dạng

phông chữ, điều kiện ánh sáng), việc xử lý trong bối cảnh video hành trình tại Việt Nam còn phải đối mặt với: chất lượng hình ảnh thay đổi liên tục, góc quay và khoảng cách khác nhau, cùng sự xuất hiện của các biến hiệu với thiết kế đa dạng và ngôn ngữ phức tạp (kết hợp tiếng Việt và tiếng Anh). Những thách thức này nhấn mạnh tầm quan trọng và tính thực tiễn của việc nghiên cứu một giải pháp hiệu quả, phù hợp với đặc thù của bài toán.

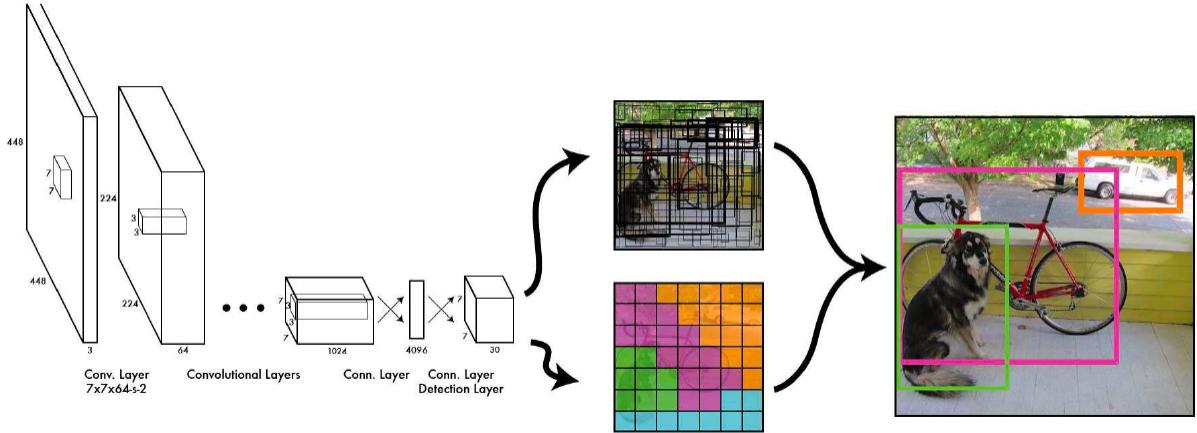
2.2 Phát hiện đối tượng

2.2.1 Cơ sở và hướng tiếp cận chung

Phát hiện đối tượng (Object Detection) là một bài toán trong lĩnh vực Thị giác Máy tính (Computer Vision), đóng vai trò trung tâm trong nhiều ứng dụng thực tiễn như giám sát an ninh, lái xe tự động, và tương tác người máy. Khác với nhiệm vụ phân loại ảnh truyền thống vốn chỉ xác định loại đối tượng xuất hiện trong toàn bộ ảnh, phát hiện đối tượng yêu cầu mô hình không chỉ nhận diện đúng loại đối tượng mà còn xác định chính xác vị trí của chúng thông qua các hộp giới hạn (bounding boxes). Thách thức của bài toán này nằm ở việc phải xử lý đồng thời nhiều đối tượng với sự đa dạng lớn về kích thước, tư thế, góc nhìn, điều kiện ánh sáng và mức độ chồng lấn giữa các đối tượng. Hình [2.1](#) minh họa tổng quan quy trình phát hiện đối tượng.

2.2.2 Các nghiên cứu liên quan

Trong những năm gần đây, cùng với sự phát triển mạnh mẽ của học sâu (deep learning), bài toán phát hiện đối tượng đã đạt được những bước tiến vượt bậc cả về độ chính xác lẫn tốc độ, thúc đẩy sự phát triển mạnh mẽ của nhiều ứng dụng thực tế như giám sát thông minh, phân tích video, robot tự hành và xe tự lái. Sự gia tăng về độ phức tạp của dữ liệu ảnh, cùng yêu cầu ngày càng cao về độ chính xác và tốc độ xử lý, đã dẫn đến sự ra đời của nhiều hướng tiếp cận khác nhau cho bài toán này. Dựa trên khảo sát tổng quan của [\[52\]](#), các phương pháp phát hiện đối tượng tiên tiến hiện nay có thể được phân loại thành ba hướng tiếp cận chính xét theo kiến trúc và quy trình xử lý:



Hình 2.1: Hình ảnh minh họa tổng quan quy trình phát hiện đối tượng. Ảnh đầu vào được xử lý qua mạng nơ-ron để trích xuất đặc trưng và dự đoán vị trí (hộp giới hạn) cùng phân loại các đối tượng (nhãn lớp) xuất hiện trong khung hình, tham khảo từ Analytics Vidhya [38].

- **Các phương pháp hai giai đoạn (Two-Stage):** Các phương pháp hai giai đoạn tiếp cận bài toán phát hiện đối tượng bằng cách tách biệt quá trình đề xuất vùng chứa đối tượng (region proposal) và quá trình phân loại định vị chi tiết. Nhóm này tiêu biểu bởi các mô hình thuộc họ R-CNN, chẳng hạn như **R-CNN** [13], **Fast R-CNN** [12] và **Faster R-CNN** [34]. Trong đó, R-CNN sử dụng các thuật toán đề xuất vùng thủ công kết hợp với CNN để trích xuất đặc trưng, trong khi Fast R-CNN cải thiện hiệu quả bằng cách chia sẻ đặc trưng toàn ảnh. Faster R-CNN tiếp tục nâng cao hiệu suất bằng cách giới thiệu mạng Region Proposal Network (RPN), cho phép học tự động các vùng đề xuất.

Nhờ khả năng tách biệt rõ ràng giữa phát hiện và phân loại, các phương pháp hai giai đoạn thường đạt độ chính xác cao, đặc biệt trong các kịch bản phức tạp, nhưng đòi hỏi chi phí tính toán lớn và tốc độ xử lý chậm.

- **Các phương pháp một giai đoạn (One-Stage):** Khác với các phương pháp hai giai đoạn, các mô hình một giai đoạn thực hiện trực tiếp việc dự đoán nhãn lớp và hộp giới hạn trong một bước duy nhất, không cần cơ chế đề xuất vùng riêng biệt. Với các đại diện nổi bật như **YOLO** [33], **SSD** [22] và **RetinaNet** [21]. YOLO tiếp cận phát hiện đối tượng như một bài toán hồi quy toàn cục, cho phép suy luận nhanh và phù hợp với các ứng dụng thời gian thực. SSD khai thác đặc trưng đa

tỷ lệ nhằm cải thiện khả năng phát hiện các đối tượng có kích thước khác nhau. RetinaNet giải quyết vấn đề mất cân bằng giữa các lớp thông qua hàm mất mát Focal Loss, giúp nâng cao độ chính xác cho các đối tượng khó phát hiện.

Nhìn chung, các phương pháp một giai đoạn (One-Stage) đạt được sự cân bằng tốt giữa tốc độ và độ chính xác, nhưng đôi khi kém ổn định hơn trong các bối cảnh có mật độ đối tượng cao hoặc chồng lấn mạnh.

- **Các phương pháp dựa trên Transformer:** Gần đây, các phương pháp dựa trên Transformer đã tạo ra một bước chuyển quan trọng trong phát hiện đối tượng bằng cách xây dựng kiến trúc end-to-end, loại bỏ các thành phần được thiết kế thủ công như anchor boxes và thuật toán Non-Maximum Suppression (NMS). Diễn hình cho hướng đi này là mô hình **DETR** [4] trong đó bài toán phát hiện đối tượng được mô hình hóa như một bài toán gán tập (set prediction) thông qua cơ chế self-attention. Các biến thể sau đó của DETR tập trung vào cải thiện tốc độ hội tụ và hiệu suất suy luận, mở ra hướng nghiên cứu mới cho các mô hình phát hiện đối tượng. Bên cạnh đó, khả năng mô hình hóa quan hệ toàn cục và thiết kế kiến trúc end-to-end của Transformer cũng đã chứng minh hiệu quả trong nhiều bài toán thị giác máy tính liên quan, bao gồm cả phân đoạn ảnh.

2.3 Phát hiện và nhận dạng văn bản ngoại cảnh (Scene Text Detection and Recognition - STDR)

2.3.1 Phát hiện văn bản ngoại cảnh (Scene Text Detection - STD)

2.3.1.1 Cơ sở và hướng tiếp cận chung

Phát hiện văn bản trong ảnh ngoại cảnh (Scene Text Detection) hướng tới mục tiêu xác định và khoanh vùng các khu vực chứa văn bản. Khác với các tác vụ phát hiện đối tượng truyền thống, phát hiện văn bản trong ảnh ngoại cảnh phải đối mặt với nhiều thách thức do sự đa dạng về hình dạng, kích thước, hướng và bộ cục của văn bản, cũng như các trường hợp văn bản bị nghiêng, cong, chồng chéo hoặc mờ. Do đó, bài toán này đòi

hồi kết hợp các kỹ thuật phát hiện đối tượng với các phương pháp chuyên biệt cho văn bản nhằm xác định chính xác và hiệu quả các vùng chứa văn bản.

2.3.1.2 Các nghiên cứu liên quan

Dựa trên các nghiên cứu khảo sát và tổng quan gần đây [15, 30, 28, 14] về phát hiện văn bản trong ảnh ngoại cảnh, các phương pháp tiên tiến hiện nay có thể được phân thành ba nhóm chính: (i) dựa trên hồi quy (regression-based), (ii) dựa trên phân đoạn (segmentation-based) và (iii) dựa trên thành phần liên thông (connected component-based).

- **Các phương pháp dựa trên hồi quy (Regression-based):** Hướng tiếp cận này giải quyết bài toán phát hiện văn bản tương tự như phát hiện đối tượng, bằng cách trực tiếp dự đoán tọa độ các vùng văn bản dưới dạng hộp chữ nhật hoặc đa giác. Nhờ kiến trúc tối ưu cho việc dự đoán tọa độ, các phương pháp này thường có tốc độ suy luận nhanh, phù hợp với các ứng dụng thời gian thực. Liao và cộng sự đã đề xuất **TextBoxes** [18], với điều chỉnh hình dạng convolutional kernel và anchor của SSD để phù hợp với tỷ lệ co giãn đa dạng của văn bản cảnh, cải thiện khả năng phát hiện văn bản đa hướng. **EAST** [50] đề xuất một pipeline hiệu quả, dự đoán trực tiếp khung bao xoay (rotated box) hoặc tứ giác (quadrangle) từ đặc trưng hình ảnh, loại bỏ sự phụ thuộc vào các bước đề xuất vùng (region proposal) phức tạp. Để xử lý văn bản hình dạng bất kỳ, Liu và cộng sự đề xuất **ABCNet** [24], sử dụng đường cong Bezier (Bezier curve) làm biểu diễn tham số hóa linh hoạt cho đường biên văn bản, cho phép mô hình hóa chính xác các văn bản cong. **FCE-Net** [51] đề xuất một cách biểu diễn khác thông qua phép nhúng chuỗi Fourier (Fourier contour embedding), giúp biểu diễn hiệu quả và tái tạo các đường biên văn bản phi chuẩn từ đầu ra hồi quy.

Tuy nhiên, một hạn chế chung của hướng tiếp cận này là sự phụ thuộc vào các bước hậu xử lý (post-processing) phức tạp để phục hồi thể hiện văn bản từ đầu ra hồi quy.

- **Các phương pháp dựa trên thành phần liên thông (Connected Component-based):** Các phương pháp này tập trung vào việc phát hiện và nhóm các thành phần ảnh có đặc trưng tương đồng (như màu sắc, kết cấu hoặc cường độ) để hình thành các vùng văn bản hoàn chỉnh. Long và cộng sự đề xuất **TextSnake** [25], mô hình hóa văn bản cong bằng một chuỗi các hình tròn linh hoạt (các "vảy rắn") dọc theo trực trung tâm. Baek và cộng sự **CRAFT** [2] dự đoán bản đồ "affinity" giữa các ký tự thông qua học chuyển giao từ dữ liệu tổng hợp, cung cấp hướng dẫn rõ ràng cho việc nhóm. **DRRG** [45] được đề xuất bởi Zhang và cộng sự, sử dụng Mạng Tích chập Đồ thị (Graph Convolutional Network - GCN) để mô hình hóa mối quan hệ cấu trúc giữa các thành phần văn bản và thực hiện việc nhóm một cách thông minh.

Mặc dù có thể biểu diễn chính xác các văn bản cong, hiệu quả cuối cùng của hướng tiếp cận này phụ thuộc nhiều vào các thuật toán nhóm (grouping) phức tạp, điều này có thể ảnh hưởng đến tốc độ xử lý và độ ổn định tổng thể.

- **Các phương pháp dựa trên phân đoạn (Segmentation-based):** Nhóm phương pháp này xem phát hiện văn bản như một bài toán phân đoạn mức pixel, phân loại mỗi pixel là văn bản hoặc nền, sau đó suy ra các vùng văn bản từ kết quả phân đoạn. **PANet** [40] được đề xuất để tổng hợp đặc trưng đa tỷ lệ và nhóm chính xác các pixel văn bản. Liao và cộng sự đề xuất **DBNet++** [20], tích hợp differentiable binarization và Adaptive Scale Fusion nhằm giảm thiểu bước hậu xử lý và cải thiện độ chính xác. **TextPMs** [44] sử dụng nhóm bản đồ xác suất và mô hình học lặp để phục hồi văn bản cong một cách chính xác. **FAST** [5] và **KPN** [46] tập trung vào việc xử lý hiệu quả các văn bản phi chuẩn và đa tỷ lệ.

Nhìn chung, phương pháp này đặc biệt hiệu quả trong việc xử lý các văn bản có hình dạng cong, hoặc nghiêng. Tuy nhiên, nó thường đòi hỏi chi phí tính toán cao hơn so với các phương pháp dựa trên hồi quy trực tiếp.

2.3.2 Nhận dạng văn bản ngoại cảnh (Scene Text Recognition - STR)

2.3.2.1 Cơ sở và hướng tiếp cận chung

Nhận dạng văn bản trong ảnh ngoại cảnh (Scene Text Recognition) là một bài toán phức tạp trong lĩnh vực Thị giác Máy tính, liên quan đến việc đọc và xác định nội dung của văn bản xuất hiện trong các cảnh ảnh thực tế. Khác với các bài toán nhận dạng văn bản truyền thống trên tài liệu hoặc bảng biểu, văn bản ngoại cảnh thường xuất hiện trong môi trường không đồng nhất, chịu biến dạng, thay đổi lớn về ánh sáng, góc chụp, nền, phông chữ và hình thức trình bày. Mục tiêu là nhận diện chính xác nội dung của các ký tự và từ ngữ trong các vùng văn bản đã được khoanh vùng (cropped text instances), chuyển đổi từng hình ảnh văn bản riêng lẻ thành chuỗi ký tự tương ứng. Bài toán không chỉ đòi hỏi nhận diện ký tự riêng lẻ mà còn cần hiểu ngữ cảnh tổng thể của văn bản trong ảnh, bao gồm mối quan hệ giữa các ký tự, từ ngữ và bối cảnh, đặc biệt trong các điều kiện biến dạng, cong, nghiêng hoặc không chuẩn.

2.3.2.2 Các nghiên cứu liên quan

Scene Text Recognition (STR) là một lĩnh vực nghiên cứu thu hút sự quan tâm mạnh mẽ trong cộng đồng thị giác máy tính. Trong các nghiên cứu khảo sát và tổng quan gần đây [15, 30, 28, 7], bài toán nhận dạng văn bản trong ảnh ngoại cảnh có thể được chia thành 2 loại chính dựa trên nguyên lý làm việc: (i) các phương pháp dựa trên phân đoạn ký tự (segmentation-based) và (ii) các phương pháp không dựa trên phân đoạn (segmentation-free).

- **Các phương pháp dựa trên phân đoạn ký tự (Segmentation-based):** Nhóm phương pháp này tiếp cận bài toán STR bằng cách dự đoán nhãn mức pixel cho từng ký tự hoặc thành phần ký tự, sau đó thực hiện nhận dạng dựa trên kết quả phân đoạn. **MaskTextSpotter** [27] được đề xuất bởi Lyu và cộng sự, sử dụng mạng phân đoạn ký tự kết hợp cơ chế spatial attention để nhận dạng văn bản hình dạng bất kỳ, khắc phục hạn chế do thiếu dữ liệu chú thích cấp ký tự. Để cải thiện độ chính xác trong bối cảnh phức tạp, Ye và cộng sự đề xuất **TextFuseNet** [42],

một kiến trúc tích hợp thông tin đa cấp (character-, word-, và global-level), giúp phân đoạn ký tự mạnh mẽ hơn.

Tuy đạt hiệu quả cao trong việc biểu diễn văn bản phi chuẩn, các phương pháp trong nhóm này thường phụ thuộc vào chất lượng của bước phân đoạn ký tự, đồng thời đòi hỏi quy trình hậu xử lý phức tạp, dẫn đến chi phí tính toán cao và khó mở rộng trong các kịch bản thực tế

- **Các phương pháp không dựa trên phân đoạn (Segmentation-free):** Nhóm phương pháp này tập trung vào việc trực tiếp ánh xạ toàn bộ vùng ảnh văn bản (word hoặc text line) thành chuỗi ký tự đầu ra, thông qua một kiến trúc encoder-decoder. Các phương pháp truyền thống trong nhóm này thường sử dụng mạng CNN để trích xuất đặc trưng thị giác, kết hợp với mô hình chuỗi như BiLSTM để nắm bắt quan hệ ngữ cảnh, và sử dụng CTC hoặc attention làm cơ chế dự đoán. Tiêu biểu là **CRNN [35]**, trong đó Shi và cộng sự kết hợp CNN, RNN và CTC loss để thực hiện nhận dạng chuỗi ký tự một cách hiệu quả.

Gần đây, các phương pháp tiên tiến dựa trên Transformer đã đạt được nhiều kết quả vượt trội. **ViTSTR [1]** được đề xuất nhằm áp dụng Vision Transformer trực tiếp cho bài toán STR, khai thác cơ chế self-attention để mô hình hóa quan hệ toàn cục trong chuỗi đặc trưng. **PARSeq [3]** tiếp tục mở rộng hướng tiếp cận này bằng cách kết hợp mô hình Transformer tự hồi quy với ngữ cảnh hai chiều, giúp cải thiện độ chính xác trong nhận dạng chuỗi dài và phức tạp. Bên cạnh đó, một số nghiên cứu gần đây như **CDistNet [49]**, **SMTR [8]** và **SVTRv2 [10]** tập trung vào việc thiết kế kiến trúc hiệu quả hơn cho STR, thông qua cải tiến cơ chế trích xuất đặc trưng, mô hình hóa chuỗi hoặc tối ưu hóa cấu trúc Transformer, nhằm cân bằng giữa độ chính xác và chi phí tính toán.

Nhìn chung, hướng tiếp cận này có kiến trúc tương đối gọn nhẹ và khả năng mở rộng tốt nhờ không phụ thuộc vào chú thích ký tự chi tiết. Tuy nhiên, việc không sử dụng phân đoạn tường minh khiến các phương pháp này gặp hạn chế khi xử lý văn bản có hình dạng phức tạp, cong hoặc biến dạng mạnh.



Hình 2.2: Hình ảnh minh họa quá trình nhận dạng văn bản ngoại cảnh đầu-cuối (End-to-End Scene Text Recognition), tham khảo từ [23].

2.3.3 Nhận dạng văn bản ngoại cảnh đầu–cuối (End-to-End Scene Text Recognition)

2.3.3.1 Cơ sở và hướng tiếp cận chung

Nhận dạng văn bản ngoại cảnh đầu–cuối (End-to-End Scene Text Recognition) hướng tới mục tiêu giải quyết đồng thời cả hai bài toán phát hiện văn bản (text detection) và nhận dạng văn bản (text recognition) trong một pipeline thống nhất, thay vì xử lý chúng như hai tác vụ tách biệt. Khác với các hệ thống truyền thống theo pipeline tuần tự, trong đó kết quả phát hiện văn bản được sử dụng làm đầu vào cho bước nhận dạng, các phương pháp end-to-end tìm cách tối ưu hóa toàn bộ quá trình từ ảnh đầu vào đến chuỗi ký tự đầu ra một cách thống nhất. Quy trình tổng quát của hướng tiếp cận này được minh họa trong Hình 2.2

Cách tiếp cận này cho phép mô hình học được mối quan hệ chặt chẽ giữa vị trí, hình dạng và nội dung của văn bản trong ảnh, từ đó giảm thiểu sự phụ thuộc vào các bước trung gian và hạn chế sai lệch lan truyền giữa các giai đoạn. Do đó, End-to-End Scene Text Recognition được xem là hướng tiếp cận hiệu quả cho các ứng dụng thực tế đòi hỏi độ chính xác cao và quy trình xử lý gọn nhẹ.

2.3.3.2 Các nghiên cứu liên quan

Trong bối cảnh ngày càng nhiều ứng dụng thực tế yêu cầu trích xuất thông tin văn bản trực tiếp từ ảnh, chẳng hạn như dịch tự động, phân tích nội dung hình ảnh hay hỗ trợ người dùng trong môi trường thông minh, việc xử lý phát hiện và nhận dạng văn bản

trong một pipeline thống nhất ngày càng trở nên cần thiết. Chính vì vậy, nhiều nghiên cứu gần đây tập trung vào bài toán nhận dạng văn bản ngoại cảnh đầu-cuối (End-to-End Scene Text Recognition), nhằm đồng thời giải quyết hai nhiệm vụ phát hiện và nhận dạng văn bản trong cùng một pipeline.

Theo các nghiên cứu khảo sát và tổng quan gần đây [15, 30, 28, 7], bài toán nhận dạng văn bản ngoại cảnh có thể được chia thành hai hướng tiếp cận chính: (i) các phương pháp hai giai đoạn (two-stage scene text spotters) và (ii) các phương pháp một giai đoạn (one-stage scene text spotters).

- **Các phương pháp hai giai đoạn (Two-Stage Scene Text Spotters):** Các phương pháp thuộc nhóm này tiếp cận bài toán text spotting bằng cách kết hợp một mô-đun phát hiện văn bản và một mô-đun nhận dạng văn bản riêng biệt trong một pipeline xử lý tuần tự. Tiêu biểu cho hướng tiếp cận này, **TextBoxes** [18] sử dụng bộ phát hiện dựa trên SSD và bộ nhận dạng CRNN, đặt nền móng cho kiến trúc hai giai đoạn trong bài toán text spotting. Tuy nhiên, việc tối ưu tách biệt hai mô-đun có thể gây lan truyền lỗi do thiếu sự phối hợp giữa phát hiện và nhận dạng, từ đó làm suy giảm độ chính xác tổng thể. Gần đây, **MaskTextSpotter** [27] sử dụng mô-đun Region-of-Interest (RoI) để trích xuất các vùng ứng viên và đưa vào nhánh Fast R-CNN nhằm sinh bản đồ phân đoạn ngữ nghĩa, cho phép xử lý hiệu quả văn bản có hình dạng bất kỳ. Một hướng tiếp cận khác được thể hiện trong **ABCNet** [24] với việc sử dụng BezierAlign, một phép biến đổi có tham số học được giúp chuyển đổi chính xác các vùng văn bản hình dạng bất kỳ (đặc biệt là văn bản cong) thành các đặc trưng đầu vào chuẩn cho bộ nhận dạng.

Mặc dù hướng tiếp cận hai giai đoạn (Two-Stage) mang lại hiệu quả cao nhờ khả năng kết hợp các mô-đun phát hiện và nhận dạng mạnh mẽ trong cùng một hệ thống, cấu trúc xử lý tuần tự và sự phụ thuộc vào các bước trung gian như như đề xuất vùng (Region of Interest - RoI) khiến các phương pháp này gặp thách thức về hiệu suất và khả năng mở rộng, đặc biệt trong các ứng dụng yêu cầu xử lý nhanh và gọn.

- **Các phương pháp một giai đoạn (One-Stage Scene Text Spotters):** Nhằm giảm

thiểu sự phụ thuộc vào các bước trung gian như đề xuất vùng (Region of Interest - RoI) và đơn giản hóa pipeline xử lý, các phương pháp một giai đoạn tích hợp trực tiếp phát hiện và nhận dạng văn bản vào một mạng duy nhất, cho phép dự đoán văn bản theo cách đầu-cuối (end-to-end). **PGNet** [39] dự đoán văn bản một cách trực tiếp thông qua việc học chuỗi các điểm trung tâm. Trong khi đó, **DeepSolo** [43], lấy cảm hứng từ ABCNet, đề xuất cơ chế biểu diễn đường cong trung tâm Bezier đơn giản hơn kết hợp với công thức truy vấn mới, cho phép phân loại ký tự chỉ thông qua phép chiếu tuyến tính từ các đặc trưng truy vấn.

Bên cạnh đó, một số nghiên cứu gần đây như **TESTR** [47], **UNITS** [16] và **DNTTextSpotter** [31] tập trung vào việc thiết kế kiến trúc thống nhất cho bài toán text spotting, thông qua khai thác Transformer, cơ chế truy vấn hoặc biểu diễn đặc trưng linh hoạt, nhằm cải thiện khả năng học đầu-cuối và giảm sự phụ thuộc vào các bước trung gian.

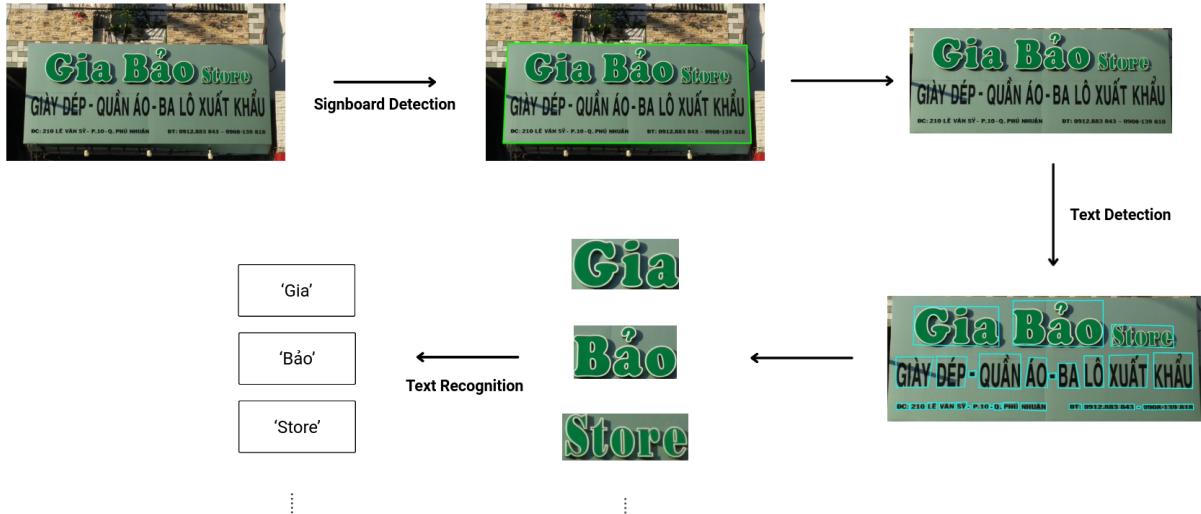
Như vậy, hướng tiếp cận một giai đoạn (One-Stage) giúp đơn giản hóa kiến trúc và giảm độ trễ suy luận nhờ loại bỏ các bước xử lý trung gian. Song, việc học đồng thời hai nhiệm vụ trong một kiến trúc duy nhất khiến mô hình dễ gặp khó khăn trong việc cân bằng giữa độ chính xác phát hiện và khả năng nhận dạng, đặc biệt trong các điều kiện dữ liệu phức tạp.

Chương 3

XÂY DỰNG QUY TRÌNH XỬ LÝ ĐẦU-CUỐI CHO PHÁT HIỆN VÀ NHẬN DẠNG VĂN BẢN TRÊN BIỂN HIỆU

3.1 Tổng quan quy trình xử lý đầu-cuối

Trong những năm gần đây, việc khai thác thông tin từ văn bản trên biển hiệu có ý nghĩa quan trọng trong việc xây dựng hệ sinh thái dữ liệu cho các ứng dụng dựa trên vị trí và quản lý đô thị. Xuất phát từ mục tiêu đó, khóa luận đề xuất một quy trình xử lý đầu-cuối (pipeline end-to-end) được thiết kế theo ba giai đoạn chính: (i) **Phát hiện biển hiệu**, (ii) **Phát hiện văn bản trong vùng biển hiệu**, và (iii) **Nhận dạng nội dung văn bản**. Theo đó, quy trình xử lý (pipeline) được minh họa trong Hình 3.1, trong đó quá trình xử lý bắt đầu bằng việc phát hiện và trích xuất vùng biển hiệu từ ảnh đầu vào. Từ các vùng biển hiệu đã được cắt, hệ thống tiến hành phát hiện các vùng văn bản tương ứng, trước khi thực hiện nhận dạng nội dung văn bản từ các vùng đã được xác định. Dựa trên quy trình xử lý (pipeline) đề xuất, khóa luận tập trung phân tích và lựa chọn một số phương pháp tiên tiến hiện nay cho từng giai đoạn xử lý. Việc lựa chọn này được thực hiện dựa trên các nghiên cứu khảo sát gần đây trong lĩnh vực. Nội dung chi tiết cho từng giai đoạn sẽ được trình bày lần lượt trong các mục sau.



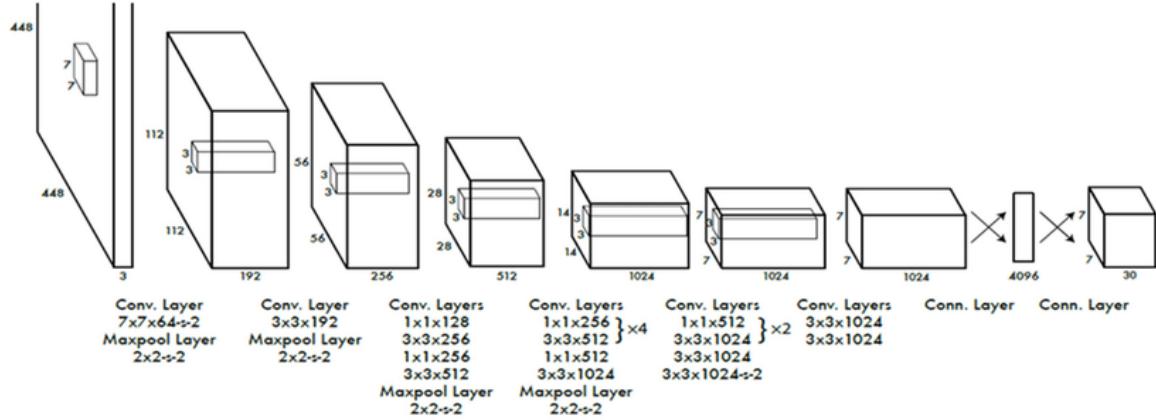
Hình 3.1: Tổng quan quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu.

3.2 Phát hiện biển hiệu

Trong giai đoạn phát hiện biển hiệu, khóa luận lựa chọn một số phương pháp tiêu biểu được đề cập trong các nghiên cứu gần đây [52] để tiến hành đánh giá thực nghiệm.

YOLO (You Only Look Once) YOLO [33] được đề xuất bởi Redmon và cộng sự, là đại diện tiêu biểu cho hướng tiếp cận một giai đoạn (one-stage) trong bài toán phát hiện đối tượng. Khác với các phương pháp hai giai đoạn, YOLO tiếp cận bài toán như một bài toán hồi quy toàn cục, trong đó mô hình dự đoán trực tiếp các hộp giới hạn (bounding boxes) cùng xác suất lớp (class probabilities) trên toàn bộ ảnh đầu vào. Kiến trúc tổng quát của YOLO được minh họa trong Hình 3.2, bao gồm ba thành phần chính: backbone dùng để trích xuất đặc trưng, neck nhằm kết hợp đặc trưng đa tỉ lệ, và detection head để thực hiện dự đoán hộp bao và nhãn phân loại.

Trải qua nhiều phiên bản phát triển, YOLO liên tục được cải tiến nhằm nâng cao độ chính xác trong khi vẫn duy trì hiệu quả tính toán. Do đó, khóa luận lựa chọn một số phiên bản YOLO gần đây, chẳng hạn như YOLOv8 và YOLOv11 để đưa vào thực nghiệm, qua đó đánh giá hiệu quả của phương pháp trong giai đoạn phát hiện biển hiệu. Bên cạnh đó, các biến thể YOLO hỗ trợ phát hiện hộp xoay (Oriented Bounding Box –

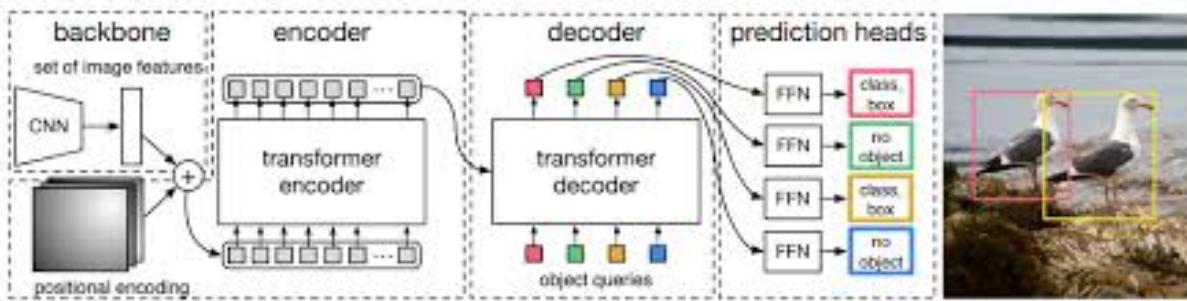


Hình 3.2: Kiến trúc tổng quan của YOLO [33]

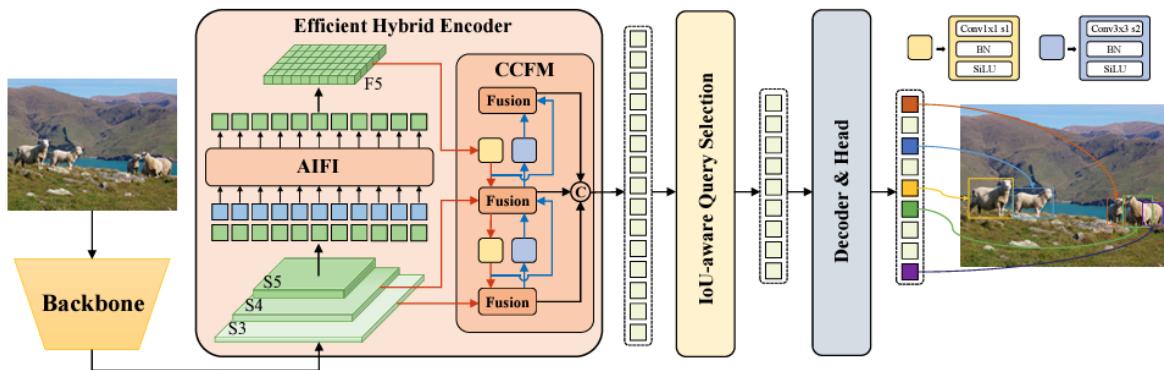
OBB) cũng được đưa vào đánh giá, nhằm xử lý hiệu quả hơn các trường hợp biến hiệu có hướng nghiêng hoặc hình dạng không song song với trực ảnh.

DETR (DEtection TRansformer) DETR [4] được đề xuất bởi Carion và cộng sự, là mô hình phát hiện đối tượng đầu tiên hoàn toàn dựa trên kiến trúc Transformer. Khác với các phương pháp dựa trên anchor truyền thống, DETR tiếp cận bài toán theo hướng dự đoán tập hợp (set prediction), trong đó mỗi đối tượng được ánh xạ trực tiếp thành một phần tử trong tập đầu ra thông qua cơ chế tự chú ý (self-attention). Cách tiếp cận này cho phép mô hình khai thác quan hệ ngữ cảnh trên toàn bộ ảnh và loại bỏ các bước hậu xử lý phức tạp như Non-Maximum Suppression (NMS). Hình 3.3 minh họa kiến trúc DETR, trong đó sử dụng CNN backbone để trích xuất đặc trưng, sau đó Transformer encoder và decoder phối hợp mô hình hóa ngữ cảnh toàn cục và sinh ra tập hợp các dự đoán cuối cùng thông qua các object queries học được.

Trong bối cảnh phát hiện biến hiệu, DETR được lựa chọn như một phương pháp đại diện cho hướng tiếp cận dựa trên Transformer nhằm đánh giá khả năng khai thác ngữ cảnh toàn cục. Đặc biệt, cơ chế dự đoán tập hợp của DETR giúp giảm thiểu sự phụ thuộc vào các giả định hình học cục bộ, từ đó phù hợp với các trường hợp biến hiệu có bố cục đa dạng.



Hình 3.3: Kiến trúc tổng quan của DETR [4]



Hình 3.4: Kiến trúc tổng quan của RT-DETR, được sử dụng trong RTDETRv2 [48]

RT-DETRv2 Dựa trên ý tưởng tiếp cận end-to-end của DETR cho bài toán phát hiện đối tượng, Zhao và cộng sự giới thiệu RT-DETRv2 [26] như một phiên bản cải tiến của RT-DETR [48], với mục tiêu tối ưu hóa hiệu suất thời gian thực trong khi vẫn duy trì độ chính xác cao. Mô hình này giữ nguyên ưu điểm loại bỏ các bước hậu xử lý như Non-Maximum Suppression (NMS), đồng thời được tăng cường bằng các cơ chế tối ưu nhằm cân bằng hiệu quả giữa tốc độ suy luận và chất lượng dự đoán. Kiến trúc của RTDETRv2, minh họa trong Hình 3.4, dựa trên thiết kế RT-DETR gốc, nổi bật với: (i) hybrid encoder kết hợp ưu điểm của CNN trong trích xuất đặc trưng hiệu quả và Transformer trong mô hình hóa ngữ cảnh toàn cục, (ii) cơ chế lựa chọn truy vấn thích ứng giúp giảm số lượng truy vấn không cần thiết, qua đó cải thiện tốc độ suy luận mà ít ảnh hưởng đến độ chính xác.

Trên cơ sở đó, RTDETRv2 được lựa chọn như một phương pháp đại diện cho hướng tiếp cận Transformer tối ưu hóa cho thời gian thực, đặc biệt phù hợp với các trường hợp

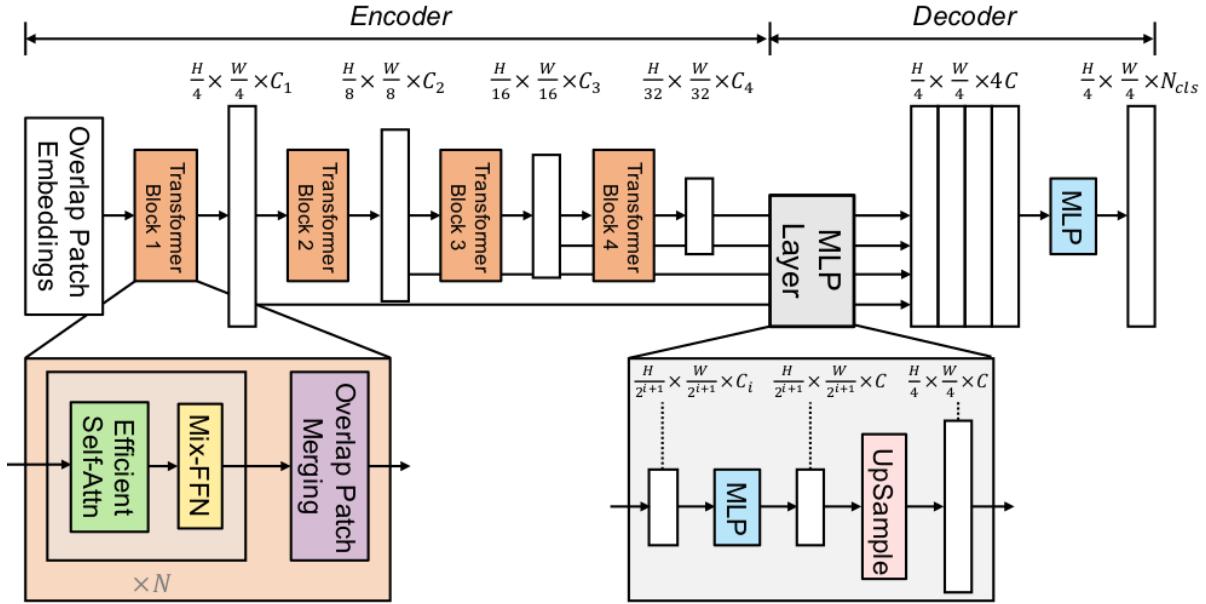
yêu cầu tốc độ xử lý cao như phân tích video giao thông hoặc cảnh đường phố.

Bên cạnh các phương pháp phát hiện trực tiếp dựa trên bounding box, nhằm mở rộng góc nhìn đánh giá, khóa luận xem xét thêm một hướng tiếp cận gián tiếp thông qua bài toán phân đoạn ngữ nghĩa (semantic segmentation). Theo hướng tiếp cận này, đối tượng được phân đoạn ở mức điểm ảnh, từ đó suy ra các vùng bao hình học phục vụ cho bài toán phát hiện biển hiệu.

Trong bối cảnh đó, theo nghiên cứu khảo sát gần đây [36], các kiến trúc dựa trên Transformer đã trở thành một hướng tiếp cận quan trọng và được quan tâm rộng rãi trong bài toán phân đoạn ảnh, đặc biệt là phân đoạn ngữ nghĩa (semantic segmentation). Nhờ khả năng mô hình hóa ngữ cảnh toàn cục thông qua cơ chế tự chú ý (self-attention), các mô hình này cho thấy hiệu quả nổi bật trong việc xử lý các trường hợp phức tạp với sự đa dạng lớn về hình dạng và bô cục của đối tượng.

SegFormer SegFormer [41], được giới thiệu bởi Xie và cộng sự, là một kiến trúc phân đoạn ngữ nghĩa hiệu quả, kết hợp encoder Transformer phân cấp (hierarchical) và decoder MLP nhẹ, được minh họa trong Hình 3.5. Thiết kế này cho phép mô hình khai thác ngữ cảnh toàn cục ở nhiều tỷ lệ, đồng thời duy trì hiệu suất tính toán cao nhờ decoder đơn giản. Chính sự cân bằng giữa độ chính xác và tốc độ này khiến SegFormer trở thành một lựa chọn phù hợp để đánh giá hiệu quả của phân đoạn ngữ nghĩa trong việc phát hiện các biển hiệu, đặc biệt đối với các biển hiệu xuất hiện ở nhiều góc nghiêng khác nhau.

Mask2Former Mask2Former [6] được đề xuất bởi Cheng và cộng sự, đại diện cho một hướng tiếp cận phân đoạn thống nhất (unified framework) dựa trên Transformer, có khả năng xử lý linh hoạt nhiều bài toán phân đoạn khác nhau như phân đoạn ngữ nghĩa (semantic segmentation), phân đoạn theo thể hiện (instance segmentation) và phân đoạn toàn cảnh (panoptic segmentation). Kiến trúc của Mask2Former được trình bày chi tiết trong Hình 3.6, áp dụng cơ chế chú ý có mặt nạ (masked attention), trong đó mỗi truy vấn tập trung vào các vùng đặc trưng liên quan đến mặt nạ (mask) dự đoán, thay vì toàn bộ không gian ảnh. Cách tiếp cận này giúp mô hình cải thiện khả năng biểu diễn hình



Hình 3.5: Kiến trúc tổng quan của SegFormer [41]

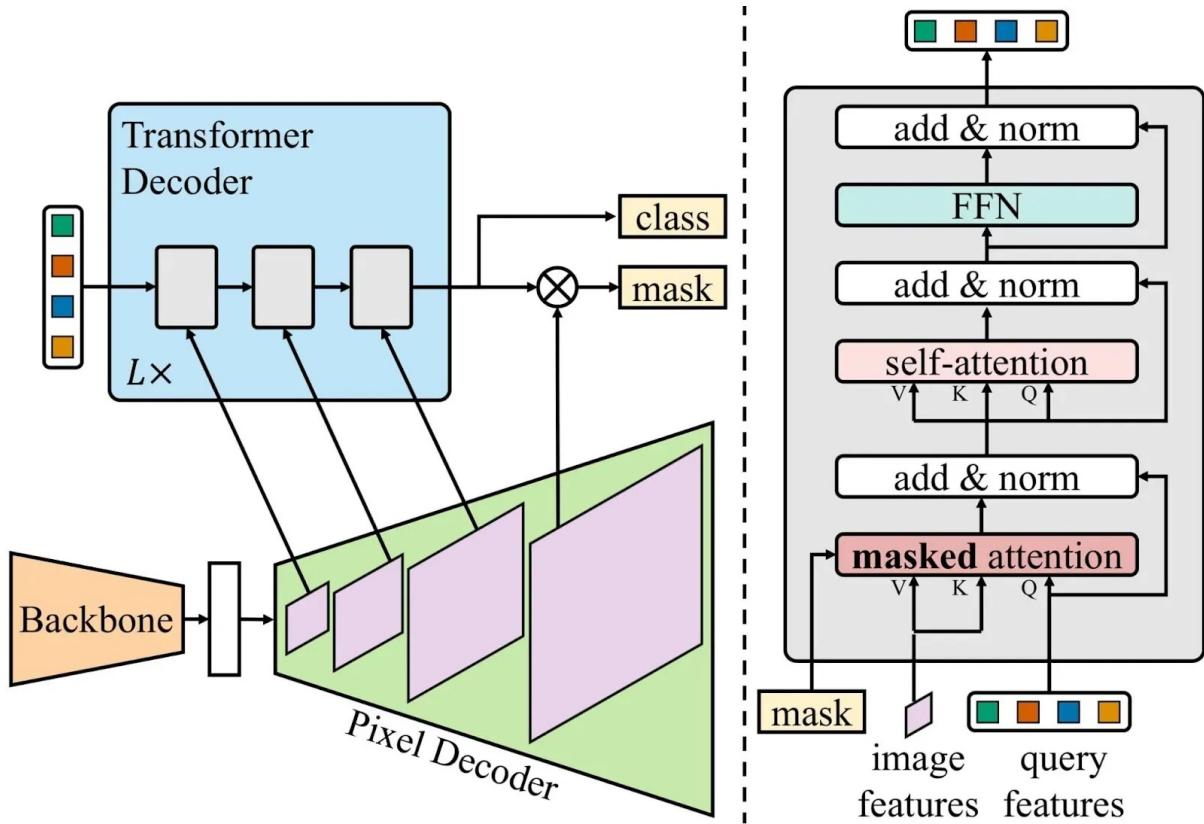
dạng và ranh giới chi tiết của các đối tượng, đặc biệt hiệu quả trong các trường hợp đối tượng chồng lấn hoặc có cấu trúc hình học phức tạp. Nhờ đó, Mask2Former được lựa chọn để đánh giá khả năng xử lý các biến hiệu trong những trường hợp bị chồng lấn.

3.3 Phát hiện và nhận dạng văn bản trên biến hiệu

3.3.1 Phát hiện văn bản

Trên cơ sở các vùng biến hiệu đã được xác định, hệ thống tiếp tục với nhiệm vụ phát hiện văn bản trên biến hiệu, được xem xét dưới góc độ của bài toán Scene Text Detection (STD). Để đánh giá hiệu quả, các phương pháp tiên tiến này được lựa chọn dựa trên các nghiên cứu khảo sát gần đây [15, 30, 28].

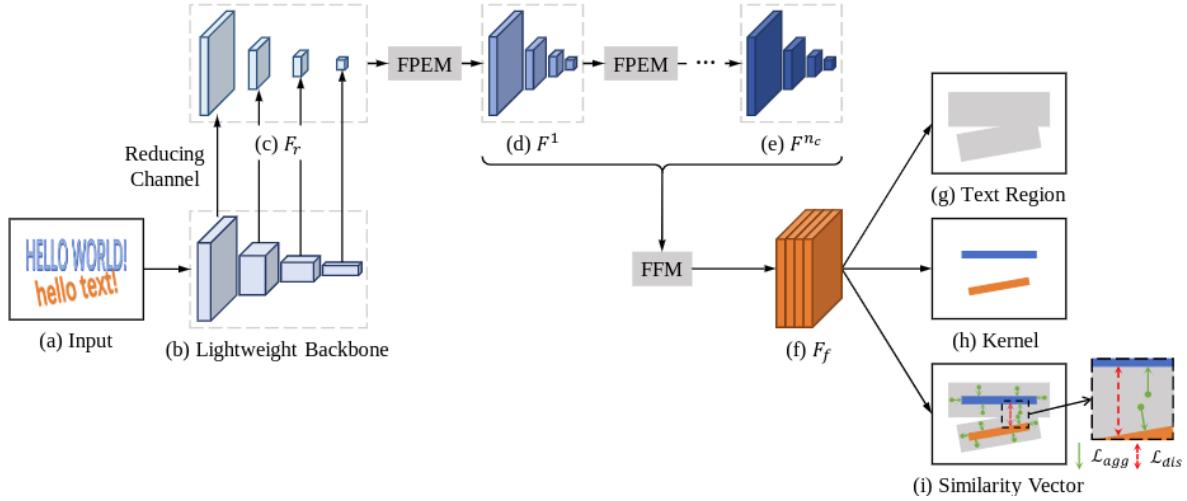
PANet PANet [40], được đề xuất bởi Liu và cộng sự, là một kiến trúc phát hiện văn bản hiệu quả dựa trên nguyên tắc phân đoạn. Mô hình bao gồm hai thành phần chính: Feature Pyramid Enhancement Module (FPEM) để tạo bản đồ đặc trưng đa tỷ lệ, và Feature Fusion Module (FFM) để tổng hợp các đặc trưng này. Kiến trúc của PANet được minh họa trong Hình 3.7. Nhờ khả năng tập hợp các pixel văn bản thành các thể



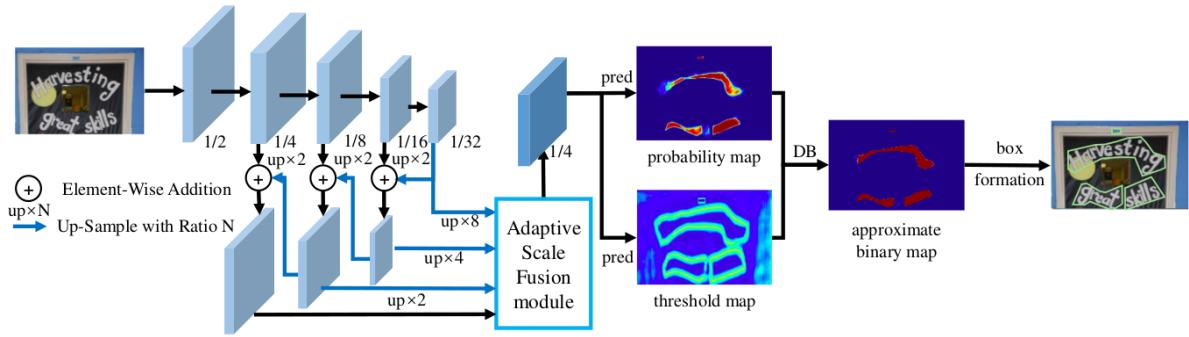
Hình 3.6: Kiến trúc tổng quan của Mask2Former [6]

hiện tương ứng trên bản đồ đặc trưng cuối cùng, PANet có thể phát hiện văn bản chính xác mà vẫn duy trì hiệu suất tính toán cao, phù hợp với bài toán phát hiện văn bản trên biển hiệu với nhiều kích thước và hướng khác nhau.

DBNet++ DBNet++ [20], được đề xuất bởi Liao và cộng sự, là phiên bản cải tiến của DBNet [19], được thiết kế để phát hiện văn bản trong ảnh ngoại cảnh với độ chính xác cao và ổn định. Mô hình tích hợp cơ chế differentiable binarization (DB) trực tiếp vào mạng phân đoạn, giúp tạo mặt nạ (mask) văn bản chính xác và giảm đáng kể các bước hậu xử lý. Bên cạnh đó, module Adaptive Scale Fusion (ASF) được áp dụng để hợp nhất các đặc trưng đa tỷ lệ, nâng cao khả năng phát hiện các văn bản có kích thước khác nhau. Kiến trúc của DBNet++ được minh họa trong Hình 3.8. DBNet++ được lựa chọn nhờ khả năng xử lý hiệu quả các đường biên văn bản không rõ nét, đồng thời phát hiện chính xác cả các dòng chữ lớn (tiêu đề) và nhỏ (thông tin chi tiết) thường cùng xuất hiện



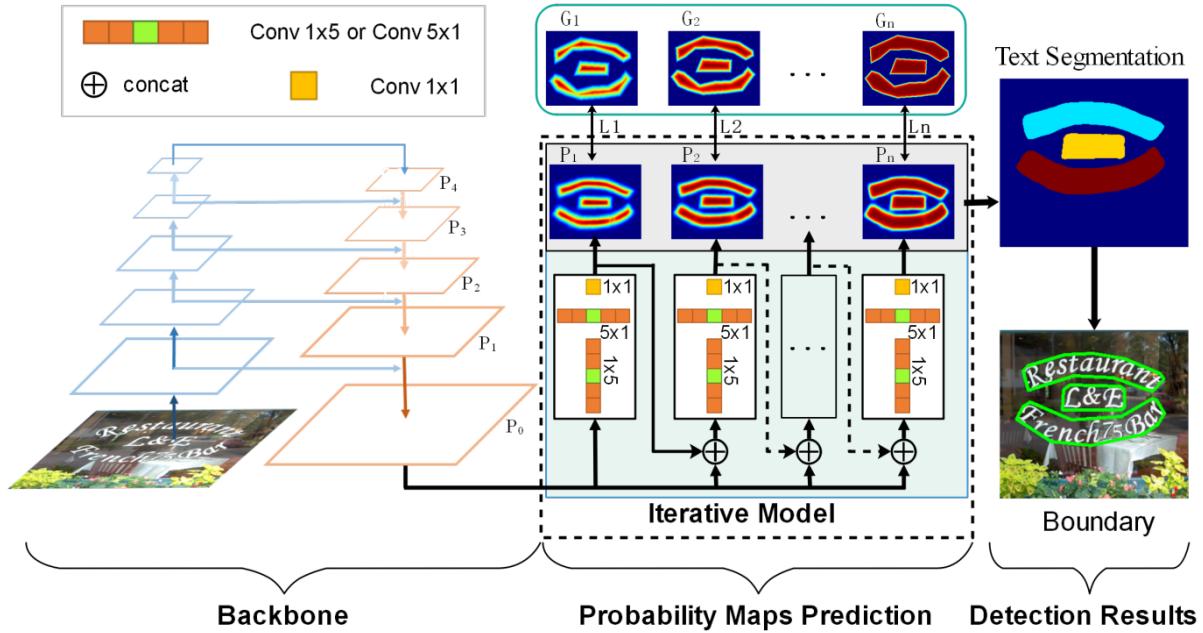
Hình 3.7: Kiến trúc tổng quan của PANet [40]



Hình 3.8: Kiến trúc tổng quan của DBNet++ [20]

trên một biển hiệu.

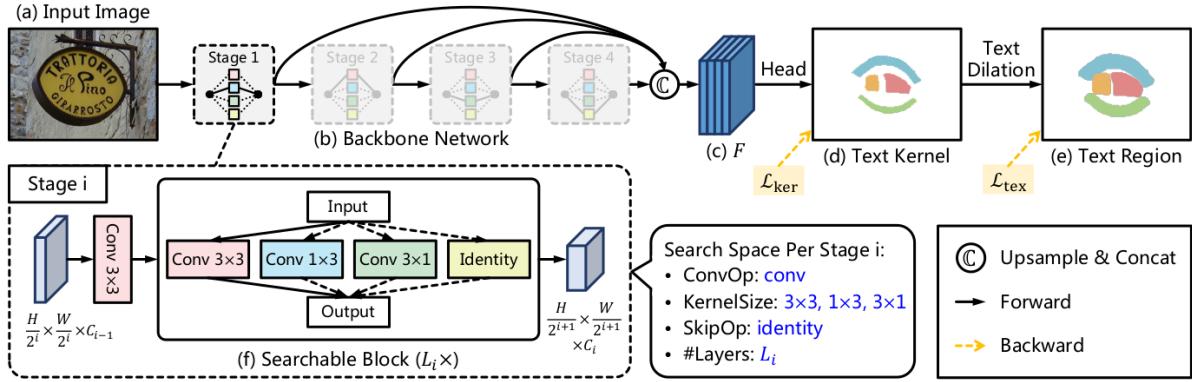
TextPMs TextPMs [44] được đề xuất bởi Zhang và cộng sự, thay vì tạo trực tiếp mặt nạ (mask) nhị phân, TextPMs dự đoán một nhóm bản đồ xác suất (probability maps) bằng cách ánh xạ khoảng cách từ pixel đến đường biên đánh dấu (annotation boundary) thành giá trị xác suất, sử dụng các hàm Sigmoid Alpha (SAF). Sau khi dự đoán nhóm bản đồ xác suất, một mô hình học lặp (iterative model) được áp dụng để tổng hợp các bản đồ này, cung cấp thông tin đầy đủ cho việc tái tạo các thể hiện văn bản. Cuối cùng, thuật toán phát triển vùng (region growth) được sử dụng để gộp các bản đồ xác suất thành các đối tượng văn bản hoàn chỉnh. Quy trình dự đoán bản đồ xác suất và phát triển vùng của TextPMs được minh họa trong Hình 3.9



Hình 3.9: Kiến trúc tổng quan của TextPMs [20]

Việc lựa chọn TextPMs dựa trên khả năng phát hiện hiệu quả các văn bản với hình dạng bất thường (như cong hoặc nghiêng), kích thước khác nhau và hướng đa dạng, đồng thời xử lý tốt các đường biên không rõ nét trên biển hiệu.

FAST Nhằm phát hiện các văn bản hình dạng tùy ý, đồng thời đảm bảo cả độ chính xác và tốc độ cao, Zhang và cộng sự đã giới thiệu FAST [5], tập trung vào việc đơn giản hóa mô hình và tối ưu hóa quá trình xử lý. Thay vì dựa vào các kiến trúc phức tạp và hậu xử lý nặng, FAST đề xuất một biểu diễn kernel tối giản (minimalist kernel) với đầu ra 1 kênh để mô hình hóa văn bản có hình dạng tùy ý, kết hợp với một quá trình hậu xử lý song song trên GPU nhằm ghép nhanh các dòng chữ với chi phí thời gian không đáng kể. Đồng thời, kiến trúc mạng của FAST được tối ưu hóa thông qua tìm kiếm kiến trúc mạng (neural architecture search) chuyên cho bài toán phát hiện văn bản, giúp trích xuất các đặc trưng mạnh mẽ và phù hợp hơn so với các mạng được thiết kế cho phân loại ảnh. Hình 3.10 cung cấp minh họa trực quan về kiến trúc tối ưu của FAST. Việc lựa chọn FAST dựa trên khả năng phát hiện hiệu quả các văn bản có hình dạng tùy ý, tối ưu cả về tốc độ lẫn độ chính xác, phù hợp với các biển hiệu xuất hiện ở nhiều kích thước,

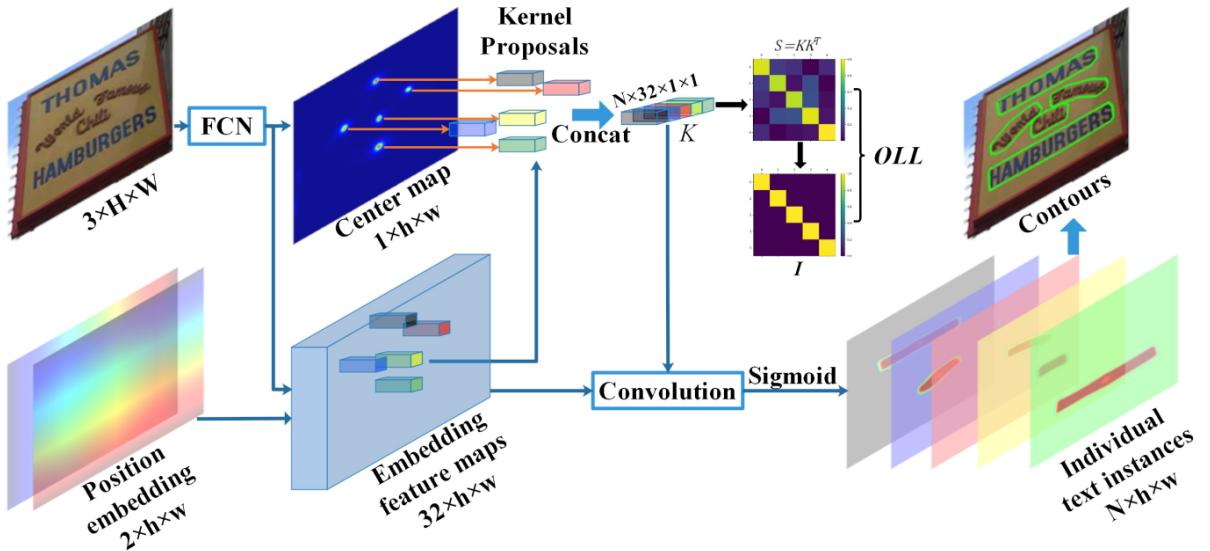


Hình 3.10: Kiến trúc tổng quan của FAST [5]

hình dạng và hướng khác nhau.

KPN Để giải quyết vấn đề tách các thể hiện văn bản liền kề trong hình ảnh ngoại cảnh, một thách thức thường gặp với các văn bản có hình dạng tùy ý. Zhang và cộng sự đề xuất KPN [46], sử dụng Kernel Proposal Network để dự đoán các bản đồ trung tâm Gaussian cho từng văn bản, từ đó trích xuất một tập hợp các kernel proposal động (dynamic convolution kernel) từ bản đồ đặc trưng embedding. Bên cạnh đó, để đảm bảo sự độc lập giữa các kernel, KPN áp dụng hàm mất mát học trực giao (orthogonal learning loss), kết hợp thông tin vị trí và thông tin ngữ nghĩa được mã hóa trong kernel. Các kernel này sau đó được áp dụng riêng rẽ lên bản đồ embedding nhằm tạo ra các bản đồ nhúng tương ứng với từng thể hiện văn bản, qua đó hỗ trợ phân tách rõ ràng các văn bản liền kề. Kiến trúc của KPN được minh họa trong Hình 3.11. Với các đặc điểm trên, KPN được lựa chọn nhờ khả năng phân tách chính xác các văn bản liền kề, đặc biệt phù hợp với các biến hiệu chứa nhiều dòng chữ gần nhau hoặc ký tự dày đặc.

YOLO (OBB) Bên cạnh các phương pháp tiên tiến cho phát hiện văn bản (Scene Text Detection - STD) đã được trình bày, khóa luận tiếp tục mở rộng đánh giá bằng cách áp dụng một số mô hình phát hiện đối tượng được giới thiệu ở Mục 3.2 (Phát hiện biến hiệu), cụ thể là phiên bản YOLOv8-OBB và YOLOv11-OBB. Do được huấn luyện ban đầu trên dữ liệu đối tượng tổng quát (general object), các mô hình này cần được fine-tune trên tập dữ liệu văn bản chuyên biệt. Việc đánh giá này nhằm xác định tính khả



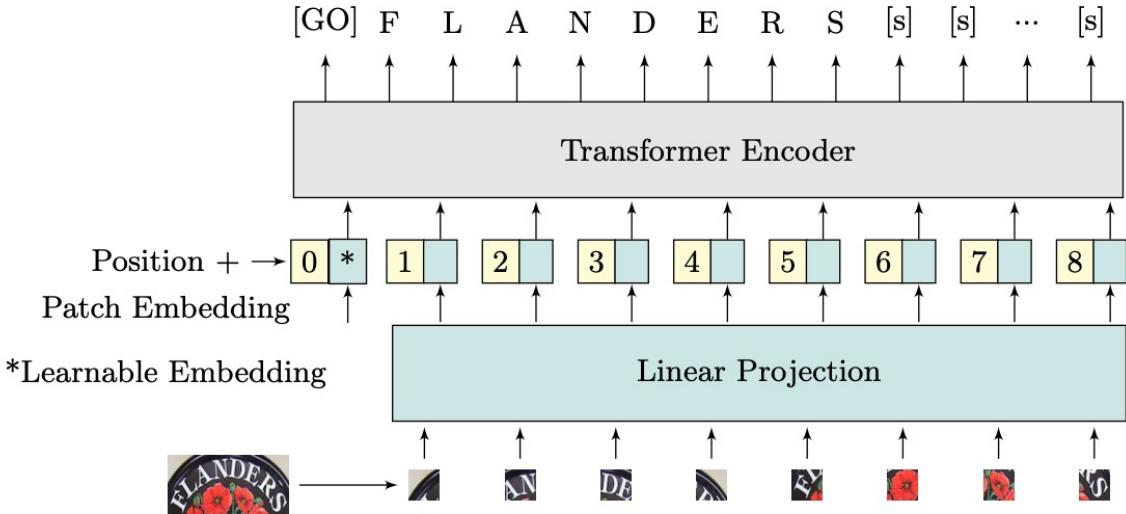
Hình 3.11: Kiến trúc tổng quan của KPN [46]

thi và hiệu quả của các kiến trúc phát hiện đối tượng khi chuyển giao (transfer) sang bài toán phát hiện văn bản, đặc biệt trong việc xử lý các dòng chữ nghiêng và có kích thước nhỏ trên biển hiệu.

3.3.2 Nhận dạng văn bản

Sau khi xác định các vùng chứa văn bản trên biển hiệu, hệ thống tiếp tục với giai đoạn nhận dạng nội dung văn bản. Giai đoạn này được tiếp cận như một bài toán Nhận dạng văn bản trong ảnh ngoại cảnh (Scene Text Recognition - STR), với mục tiêu chuyển đổi các vùng văn bản đã được phát hiện thành chuỗi ký tự tương ứng. Trên cùng cơ sở tiếp cận như giai đoạn phát hiện văn bản, khóa luận lựa chọn một số phương pháp STR tiên tiến hiện nay để tiến hành thực nghiệm và đánh giá, dựa trên các phân loại và hướng tiếp cận được tổng hợp từ các nghiên cứu khảo sát gần đây [15, 30, 28].

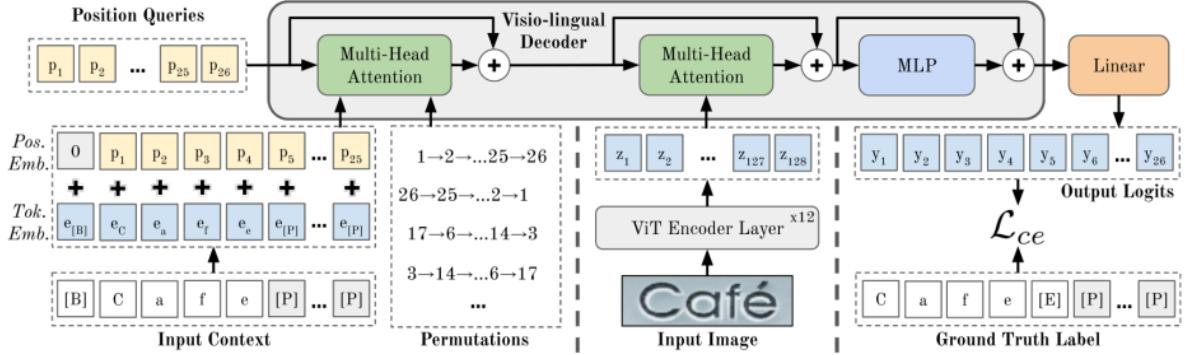
ViTSTR ViTSTR [1], được đề xuất bởi Atienza và cộng sự, đại diện cho một hướng tiếp cận đơn giản và hiệu quả khi áp dụng kiến trúc Vision Transformer cho bài toán nhận dạng văn bản trong ảnh ngoại cảnh. Mô hình sử dụng kiến trúc một giai đoạn, bao gồm 12 khối encoder Transformer giống nhau và không sử dụng decoder, như được



Hình 3.12: Kiến trúc tổng quan của ViTSTR [1]

minh họa trong Hình 3.12. Trong thiết kế này, việc dự đoán được thực hiện thông qua một lớp tuyến tính, ánh xạ trực tiếp các đặc trưng đã được mã hóa thành chuỗi ký tự đầu ra. Nhờ kiến trúc tối giản, ViTSTR đạt được hiệu quả tính toán cao, thể hiện qua tốc độ suy luận nhanh và số lượng tham số nhỏ, tạo ra một giải pháp nhẹ và nhanh phù hợp với giai đoạn nhận dạng văn bản trên biển hiệu. Ngoài ra, mô hình còn áp dụng các kỹ thuật tăng cường dữ liệu đa dạng nhằm cải thiện độ chính xác.

PARSeq Nhằm khắc phục những hạn chế của các mô hình ngôn ngữ tự hồi quy (autoregressive - AR) truyền thống, Bautista và cộng sự đã giới thiệu PARSeq [3]. Phương pháp này tận dụng kỹ thuật Permutation Language Modeling để huấn luyện một tập hợp các mô hình ngôn ngữ AR nội bộ có trọng số chung, qua đó cho phép kết hợp linh hoạt giữa suy luận không tự hồi quy mang tính độc lập ngữ cảnh (context-free non-AR) và suy luận tự hồi quy có xét đến ngữ cảnh chuỗi (context-aware AR). Trên cơ sở đó, PARSeq tích hợp cơ chế tinh chỉnh lặp (iterative refinement) dựa trên ngữ cảnh hai chiều nhằm khai thác hiệu quả thông tin ngữ cảnh mà không cần đến mô hình ngôn ngữ bên ngoài hay quy trình xử lý nhiều giai đoạn phức tạp. Nhờ vậy, mô hình thể hiện tính mạnh mẽ trước các văn bản có hướng và bối cảnh đa dạng, giúp nâng cao hiệu quả cho giai đoạn nhận dạng văn bản trên biển hiệu. Kiến trúc tổng quan của PARSeq được

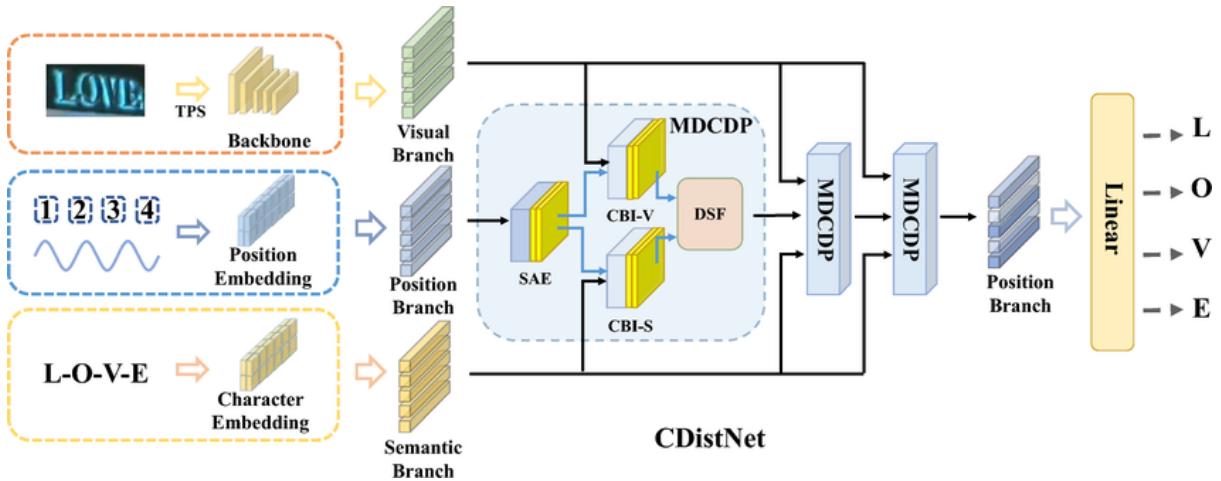


Hình 3.13: Kiến trúc tổng quan của PARSeq [3]

trình bày trong Hình 3.13.

CDistNet Để khắc phục hạn chế trong việc kết hợp thông tin thị giác và ngữ nghĩa vốn thường không được cẩn chỉnh chính xác, đặc biệt đối với các mẫu văn bản có bộ cục phức tạp hoặc biến dạng mạnh, Zheng và cộng sự đã đề xuất CDistNet [49]. Phương pháp này nhằm tăng cường mối liên kết chặt chẽ giữa hai miền đặc trưng, qua đó cải thiện khả năng căn chỉnh giữa đặc trưng và ký tự trong quá trình nhận dạng. CDistNet sử dụng một encoder gồm ba nhánh song song để trích xuất các nguồn thông tin bổ sung cho nhau, bao gồm đặc trưng thị giác từ ảnh đầu vào, đặc trưng ngữ nghĩa từ chuỗi ký tự, và embedding vị trí mô tả quan hệ không gian giữa các ký tự. Các đặc trưng này sau đó được đưa vào mô-đun Multi-Domain Character Distance Perception (MDCDP) tạo ra một embedding vị trí (positional embedding) để đồng thời truy vấn cả đặc trưng thị giác và ngữ nghĩa thông qua cơ chế chú ý chéo (cross-attention). Thông qua cơ chế này, CDistNet có khả năng trực tiếp mô hình hóa khoảng cách ký tự đa miền, bao gồm khoảng cách không gian, mối quan hệ ngữ nghĩa giữa các ký tự, cũng như sự liên kết giữa hai loại thông tin này. Cấu trúc encoder ba nhánh và mô-đun MDCDP của CDistNet được minh họa trong Hình 3.14.

Bằng cách xếp chồng nhiều mô-đun MDCDP, mô hình dần dần học được sự căn chỉnh chính xác hơn giữa vùng ảnh và ký tự, ngay cả trong các trường hợp nhận dạng khó. Nhờ đó, CDistNet hiệu quả giai đoạn nhận dạng văn bản trên biển hiệu với kiểu chữ biến dạng, xoay nghiêng hoặc bộ cục không chuẩn.

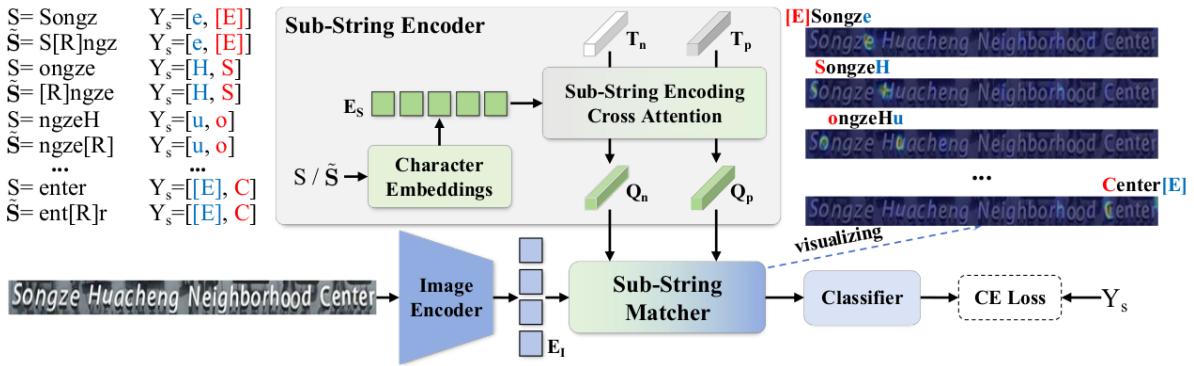


Hình 3.14: Kiến trúc tổng quan của CDistNet [49]

SMTR SMTR [8], được đề xuất bởi Du và cộng sự, áp dụng hướng tiếp cận dựa trên so khớp chuỗi con (sub-string matching) nhằm khắc phục hạn chế của các phương pháp truyền thống trong việc nhận dạng các chuỗi văn bản dài. Thay vì dự đoán toàn bộ chuỗi cùng lúc, SMTR thực hiện nhận dạng thông qua một quy trình lặp. Cụ thể, mô hình sử dụng hai mô-đun dựa trên cơ chế chú ý chéo (cross-attention), trong đó mô-đun thứ nhất mã hóa một chuỗi con gồm nhiều ký tự thành các truy vấn ngữ cảnh trước và sau, trong khi mô-đun thứ hai khai thác các truy vấn này để chú ý vào đặc trưng hình ảnh, đồng thời nhận dạng ký tự kế tiếp và ký tự liền trước của chuỗi con. Quá trình này được lặp lại nhiều lần, cho phép SMTR nhận dạng văn bản có độ dài tùy ý. Dựa trên cơ chế nhận dạng chuỗi con, SMTR có thể được huấn luyện trên các tập dữ liệu văn bản ngắn nhưng vẫn tổng quát tốt cho văn bản dài. Sơ đồ mô-đun và quy trình lặp của SMTR được trình bày trong Hình 3.15.

Ngoài ra, SMTR tích hợp chiến lược tăng cường suy luận (inference augmentation strategy) nhằm giảm thiểu sự nhầm lẫn giữa các chuỗi con tương tự. Nhờ đó, mô hình cải thiện đáng kể hiệu quả nhận dạng các chuỗi văn bản dài và phức tạp, đặc biệt phù hợp với các biến hiệu chứa nhiều từ hoặc các dòng chữ kéo dài.

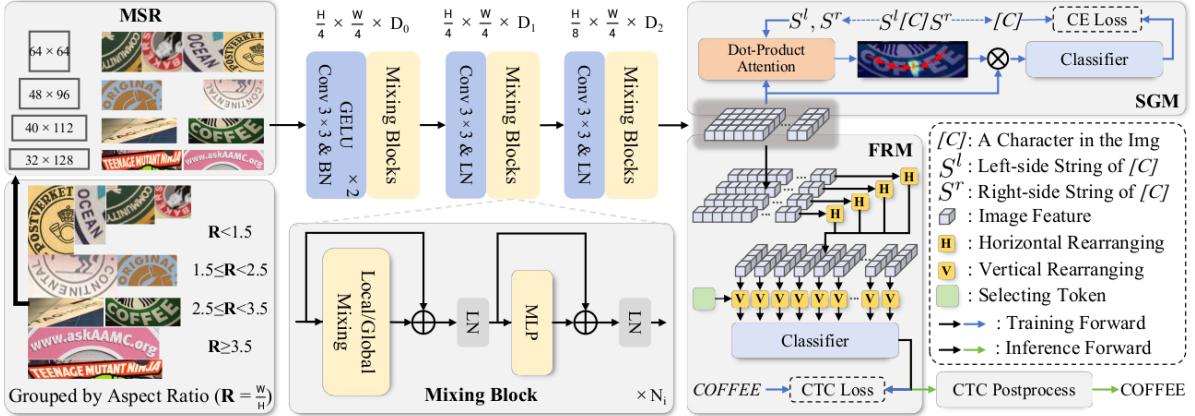
SVTRv2 Du và cộng sự đã giới thiệu SVTRv2 [10], là phiên bản mở rộng của SVTR [9], được đề xuất nhằm khắc phục những hạn chế về độ chính xác của các mô hình dựa



Hình 3.15: Kiến trúc tổng quan của SMTR [8]

trên Connectionist Temporal Classification (CTC) khi xử lý văn bản có hình dạng bất thường hoặc thiếu ngữ cảnh ngôn ngữ (linguistic missing), mặc dù các mô hình này vốn có ưu điểm về kiến trúc đơn giản và tốc độ suy luận nhanh so với các mô hình encoder-decoder. SVTRv2 áp dụng chiến lược đa kích thước (multi-size resizing) để điều chỉnh kích thước ảnh đầu vào phù hợp, tránh biến dạng nghiêm trọng, đồng thời giới thiệu mô-đun sắp xếp lại đặc trưng (feature rearrangement) để đảm bảo đặc trưng thị giác phù hợp với yêu cầu căn chỉnh của CTC. Bên cạnh đó, SVTRv2 tích hợp mô-đun định hướng ngữ nghĩa (semantic guidance module) nhằm đưa thông tin ngôn ngữ vào quá trình học đặc trưng thị giác, giúp mô hình tận dụng ngữ cảnh chuỗi để cải thiện độ chính xác. Đáng chú ý, mô-đun này chỉ được sử dụng trong giai đoạn huấn luyện và có thể loại bỏ hoàn toàn khi suy luận, do đó không làm tăng chi phí tính toán khi triển khai thực tế. Hình 3.16 minh họa kiến trúc tổng quan của SVTRv2.

Dựa trên sự kết hợp giữa hiệu quả suy luận của CTC và khả năng mô hình hóa các văn bản có hình dạng bất thường, cũng như khai thác ngữ cảnh ngôn ngữ, SVTRv2 đạt được sự cân bằng tốt giữa tốc độ và độ chính xác trong các trường hợp nhận dạng văn bản đa dạng như văn bản dài và văn bản có hình dạng phức tạp. Do đó, trong giai đoạn nhận dạng văn bản trên biển hiệu, SVTRv2 cho thấy tính phù hợp cao nhờ khả năng cân bằng giữa tốc độ suy luận và độ chính xác.



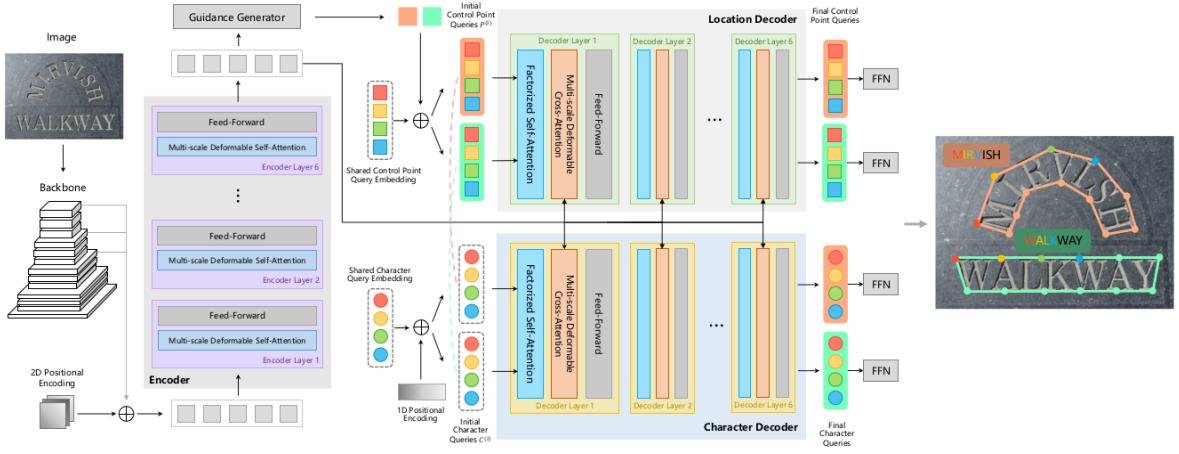
Hình 3.16: Kiến trúc tổng quan của SVTRv2 [10]

3.3.3 Phát hiện và nhận dạng văn bản đầu-cuối (end-to-end)

Mặc dù hướng tiếp cận hai giai đoạn (two-stage), trong đó phát hiện và nhận dạng văn bản được thực hiện riêng biệt, đã cho thấy hiệu quả và tính linh hoạt cao trong bài toán nhận dạng văn bản trên biển hiệu, các phương pháp một giai đoạn (one-stage) ngày càng thu hút sự quan tâm nhờ khả năng suy luận trực tiếp từ ảnh đầu vào đến chuỗi ký tự đầu ra trong một mô hình thống nhất. Do đó, bên cạnh việc xây dựng và đánh giá pipeline hai giai đoạn, khóa luận tiến hành thực nghiệm so sánh giữa hai chiến lược tiếp cận: (i) hướng tiếp cận hai giai đoạn (two-stage), với hai mô-đun riêng biệt cho phát hiện và nhận dạng; và (ii) hướng tiếp cận một giai đoạn (one-stage), sử dụng các mô hình tiên tiến như TESTR, DeepSolo, UNITS, và DNTextSpotter.

Mục tiêu của so sánh nhằm phân tích ưu nhược điểm của từng chiến lược trong bối cảnh nhận dạng văn bản trên biển hiệu, đặc biệt xét trên các khía cạnh như độ chính xác, tốc độ suy luận, mức độ phức tạp của hệ thống, đồng thời định hướng lựa chọn mô hình phù hợp để tinh chỉnh (fine-tuning) hiệu quả và tiết kiệm thời gian huấn luyện. Qua đó, khóa luận cung cấp cái nhìn tổng quan về khả năng ứng dụng thực tế của các phương pháp phát hiện và nhận dạng văn bản thống nhất (Text Spotting) hiện đại.

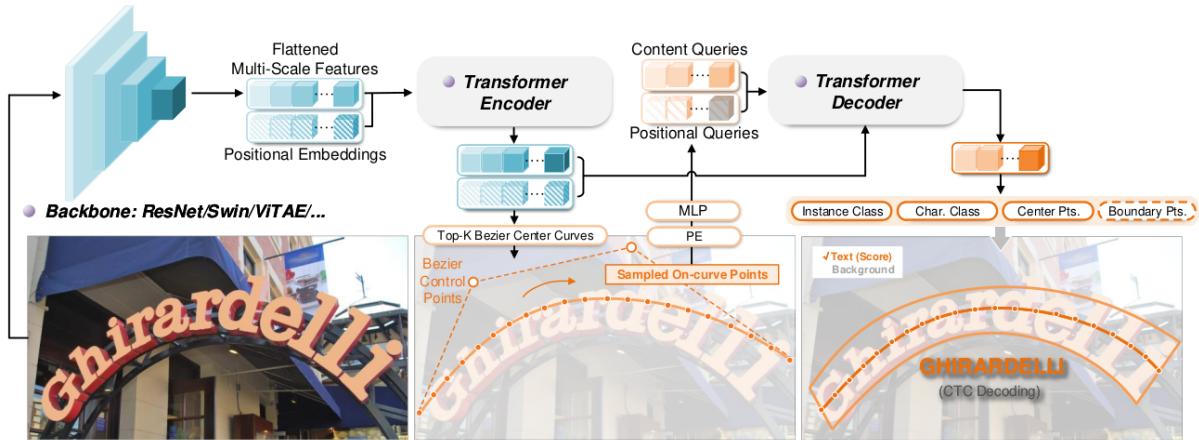
TESTR TESTR [47] được đề xuất bởi Zhang và cộng sự, nổi bật với việc áp dụng kiến trúc Transformer cho việc phát hiện và nhận dạng văn bản đầu-cuối (end-to-end) trong ảnh ngoại cảnh. Mô hình xây dựng dựa trên một bộ mã hóa (encoder) chung và hai



Hình 3.17: Sơ đồ kiến trúc của TESTR [47]

bộ giải mã (decoder) song song, lần lượt đảm nhiệm việc hồi quy các điểm điều khiển của hộp chữ (text-box control point regression) và nhận dạng ký tự. Thiết kế này giúp TESTR loại bỏ hoàn toàn các thao tác trích xuất vùng quan tâm (RoI) và các quy trình hậu xử lý phức tạp dựa trên heuristic. Trên cơ sở đó, TESTR đặc biệt hiệu quả khi xử lý các văn bản uốn cong và có hình dạng bất kỳ nhờ biểu diễn linh hoạt bằng đường cong Bezier hoặc đa giác, thay vì chỉ sử dụng hộp giới hạn hình chữ nhật truyền thống. Bên cạnh đó, quy trình phát hiện đa giác có định hướng từ hộp giới hạn (box-to-polygon detection) được đề xuất nhằm nâng cao độ chính xác định vị. Kiến trúc tổng quan của TESTR được minh họa trong Hình 3.17.

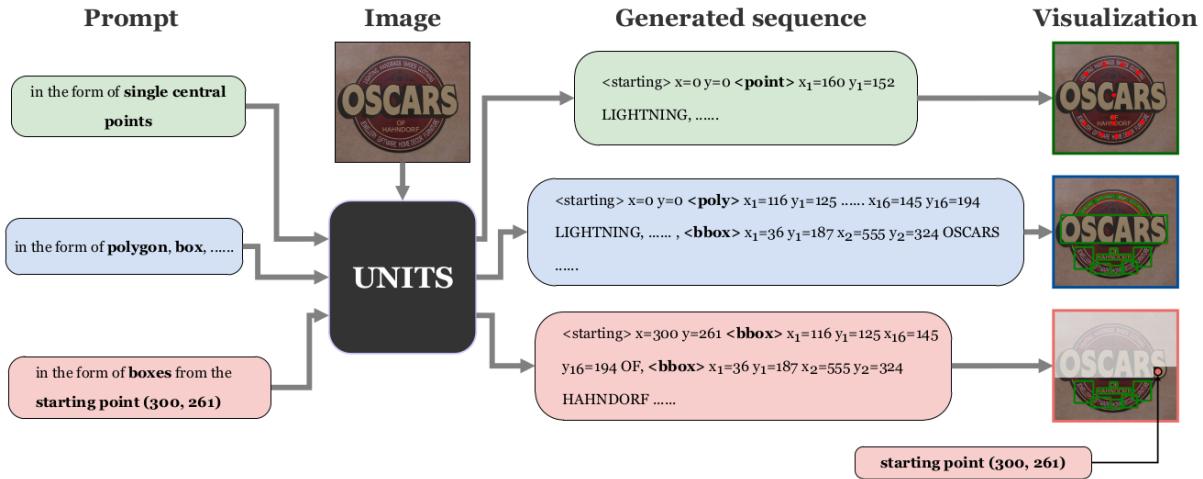
DeepSolo DeepSolo [43], được giới thiệu bởi Ye và cộng sự, nhằm giải quyết bài toán phát hiện và nhận dạng văn bản thông nhất (end-to-end) trong ảnh ngoại cảnh, nổi bật với khả năng xử lý đồng thời cả hai nhiệm vụ trong một mô hình duy nhất. Dựa trên nền tảng kiến trúc DETR, DeepSolo sử dụng một bộ giải mã (decoder) duy nhất với cơ chế Explicit Points Solo, cho phép mô hình học đồng thời để phát hiện và nhận dạng văn bản. Mỗi thực thể văn bản được biểu diễn dưới dạng chuỗi các điểm sắp xếp tự nhiên, và được mô hình hóa thông qua các truy vấn điểm có thể học được (learnable explicit point queries). Sau khi đi qua decoder, các truy vấn này mã hóa thông tin ngữ nghĩa và vị trí của văn bản, từ đó có thể giải mã để xác định đường trung tâm (center line), biên giới (boundary), kiểu chữ (script) và độ tin cậy (confidence) thông qua các đầu ra dự



Hình 3.18: Kiến trúc tổng quan của DeepSolo [43]

đoán song song đơn giản. Nhờ những đặc điểm này, DeepSolo đạt hiệu quả cao cả về độ chính xác lẫn tốc độ huấn luyện trên các bộ dữ liệu chuẩn, đồng thời cung cấp giải pháp linh hoạt, phù hợp cho bài toán nhận dạng văn bản trên biển hiệu. Sơ đồ khôi (block diagram) của DeepSolo được thể hiện trong Hình 3.18.

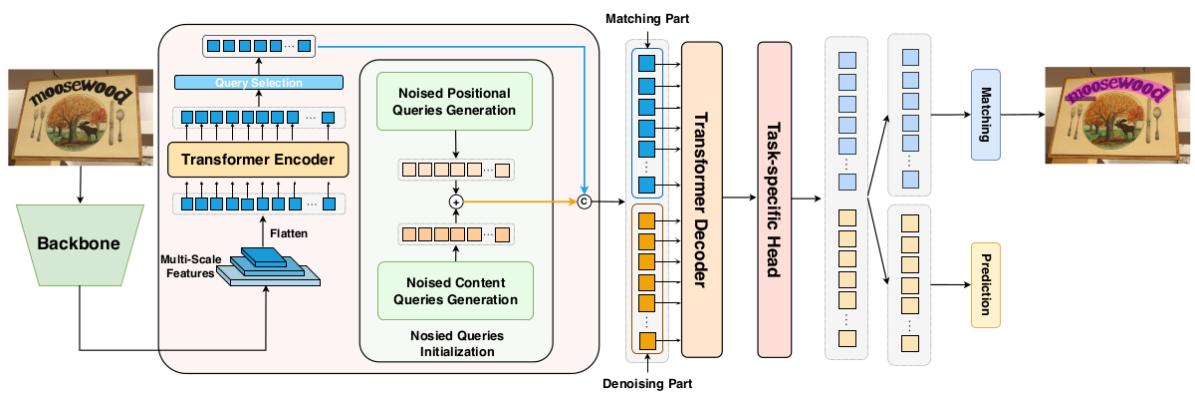
UNITS Nhằm khắc phục một số hạn chế về định dạng phát hiện và số lượng văn bản trong các mô hình tự hồi quy (auto-regressive) trước đó, Kil và cộng sự đã đề xuất UNITS (UNIfied Text Spotter) [16], nổi bật với khả năng thống nhất nhiều định dạng phát hiện, bao gồm tứ giác (quadrilateral) và đa giác (polygon), giúp mô hình xử lý văn bản có hình dạng bất kỳ. UNITS hoạt động theo cơ chế tạo chuỗi (sequence generation), trong đó thông tin của mỗi thể hiện văn bản (text instance) trong chuỗi đầu ra bao gồm token định dạng phát hiện, các token tọa độ cho việc định vị và chuỗi ký tự nhận dạng. Đặc biệt, kỹ thuật starting-point prompting được tích hợp, cho phép mô hình bắt đầu trích xuất văn bản từ một vị trí bất kỳ, từ đó có thể phát hiện nhiều thực thể văn bản vượt quá số lượng mà mô hình đã được huấn luyện. Pipeline hoạt động theo cơ chế tạo chuỗi (sequence generation) của UNITS được trình bày trong Hình 3.19. Nhờ khả năng mở rộng linh hoạt này, UNITS được lựa chọn là một giải pháp phù hợp cho giai đoạn phát hiện và nhận dạng văn bản trên biển hiệu, đặc biệt trong các trường hợp văn bản có hình dạng đa dạng và mật độ cao.



Hình 3.19: Pipeline tổng quan của UNITS [16]

DNTTextSpotter DNTTextSpotter [31], được giới thiệu bởi Qiao và cộng sự, nhằm cải thiện độ ổn định và hiệu quả huấn luyện trong các phương pháp phát hiện và nhận dạng văn bản đầu-cuối (end-to-end text spotting) dựa trên kiến trúc Transformer. Các đặc trưng đa tỉ lệ (multi-scale features) được trích xuất từ backbone và bộ mã hóa (encoder), sau đó được đưa vào một bộ giải mã (decoder) với thiết kế hai nhánh đặc biệt: (i) gồm phần ghép cặp (matching part), sử dụng các truy vấn khởi tạo ngẫu nhiên và tính toán hàm mất mát thông qua thuật toán ghép cặp đồ thị hai phía (bipartite graph matching), và (ii) phần khử nhiễu (denoising part) căn chỉnh giữa vị trí và nội dung bằng các truy vấn vị trí nhiễu (noised positional queries) và truy vấn nội dung nhiễu (noised content queries). Trong đó, các truy vấn vị trí được tạo ra từ bốn điểm điều khiển Bezier của đường trung tâm, còn các truy vấn nội dung được khởi tạo thông qua phương pháp trượt ký tự có mặt nạ (masked character sliding), đồng thời một hàm mất mát bổ sung cho việc phân loại ký tự nền được tích hợp nhằm tăng cường khả năng nhận biết ngữ cảnh. Hình 3.20 minh họa kiến trúc DNTTextSpotter.

Dựa trên cơ chế khử nhiễu và khả năng căn chỉnh vị trí-nội dung hiệu quả, DNTText Spotter chọn cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu với hình dạng đa dạng và phức tạp.



Hình 3.20: Kiến trúc tổng quan của DNTextSpotter [31]

Chương 4

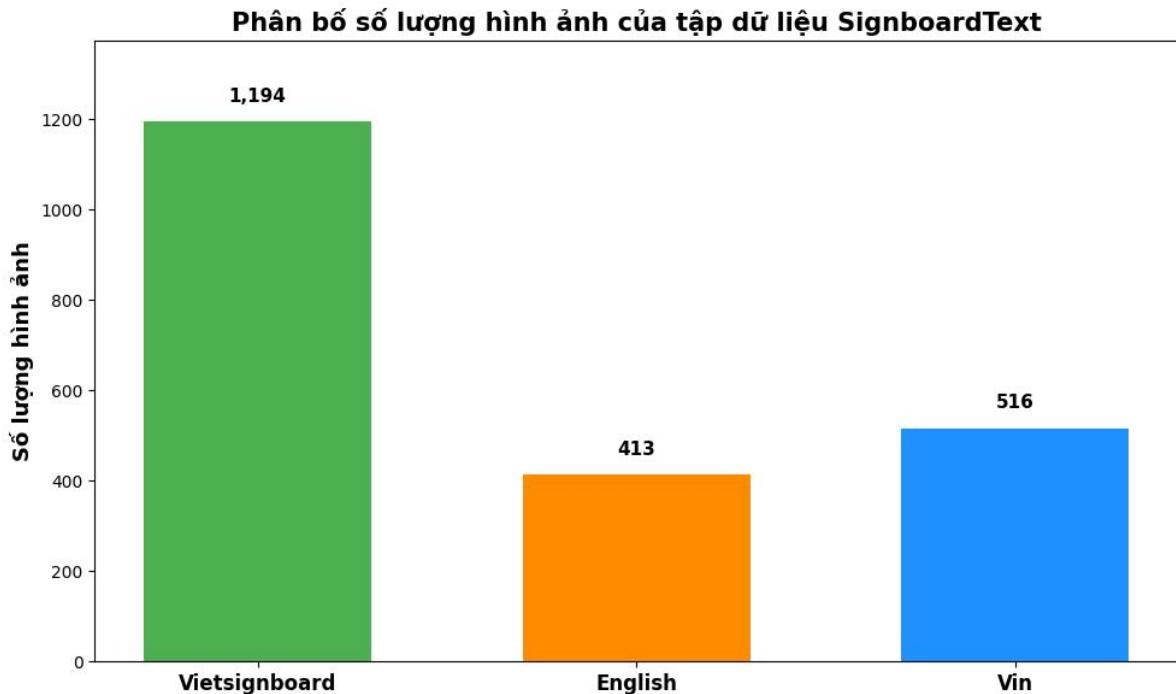
THỰC NGHIỆM VÀ ĐÁNH GIÁ

4.1 Tập dữ liệu

Để xây dựng và đánh giá hiệu quả cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video đường phố Việt Nam, khóa luận sử dụng bộ dữ liệu **SignboardText** được giới thiệu bởi [7]. Bộ dữ liệu này cung cấp một tập dữ liệu chuyên biệt cho văn bản trên biển hiệu, với các thách thức đặc thù như văn bản đa ngôn ngữ (tiếng Anh và tiếng Việt), kiểu chữ nghệ thuật, đặc biệt sự xuất hiện của các dấu thanh (tone marks) trong tiếng Việt, một yếu tố có thể ảnh hưởng đáng kể đến độ chính xác của các phương pháp hiện tại vốn thường được huấn luyện trên các ngôn ngữ không dấu phổ biến như Tiếng Anh.

Dựa trên nghiên cứu nền tảng [7], khóa luận tiến hành phân tích và thống kê cấu trúc của bộ dữ liệu SignboardText, bao gồm ba tập con chính: Vietsignboard, English và Vin. Trong đó, tập Vietsignboard đóng vai trò là tập dữ liệu chính, với 1,327 ảnh được [7] thu thập thủ công trên đường phố Việt Nam, trong khi các tập English và Vin được bổ sung với 413 và 516 ảnh, chọn lọc từ các bộ dữ liệu benchmark Total-Text, ICDAR2015 và VinText, nhằm tăng cường tính đa dạng cho bộ dữ liệu. Sự phân bố số lượng ảnh của ba tập con được minh họa trong Hình 4.1.

Về định dạng gán nhãn (annotation), Vietsignboard cung cấp nhãn ở cả hai cấp độ: cấp độ từ (word-level) với 48.638 thể hiện (instances) và cấp độ dòng (line-level) với 10.950 thể hiện (instances). Trong khi đó, các tập English và Vin chỉ được gán nhãn ở

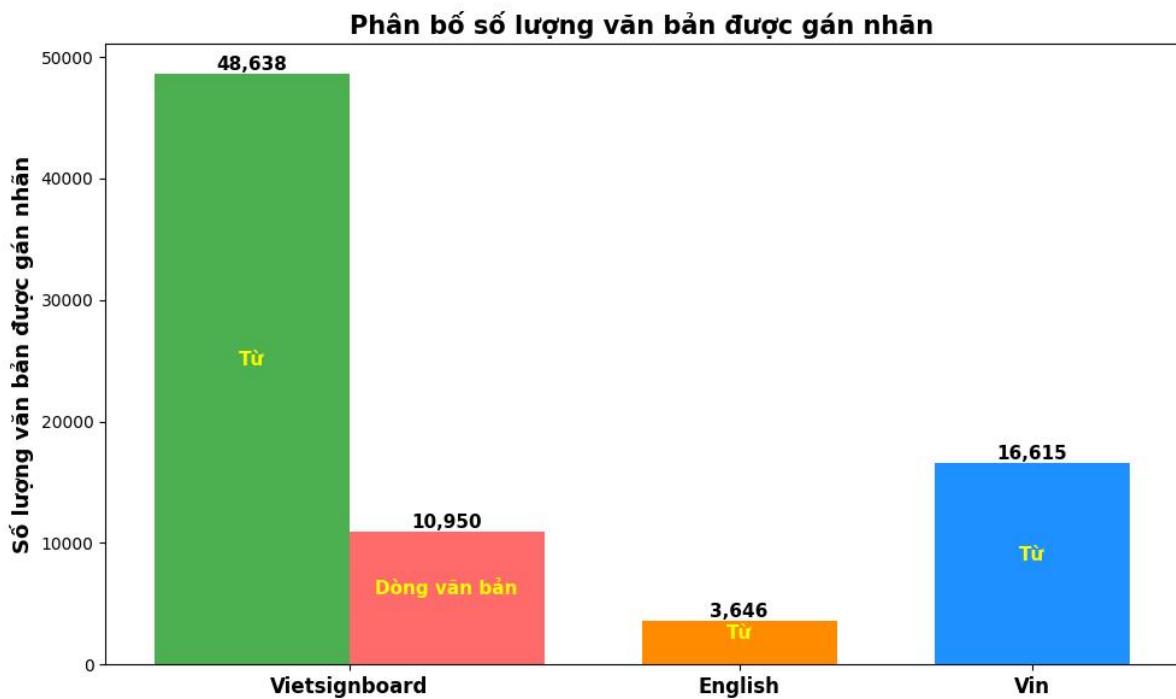


Hình 4.1: Phân bố số lượng hình ảnh trong ba tập con của SignboardText [7]

cấp độ từ (word-level) với lần lượt 3,646 và 16,615 thể hiện (instances). Phân bố chi tiết của các loại annotation này được trình bày trong Hình 4.2. Như vậy, trong bộ dữ liệu SignboardText, phần lớn văn bản được gán nhãn ở cấp độ từ (word). Đồng thời, việc kết hợp gán nhãn ở cả cấp độ từ (word) và dòng (line) trong tập Vietsignboard tạo điều kiện thuận lợi cho việc đánh giá linh hoạt hai nhiệm vụ phát hiện và nhận dạng văn bản.

Theo phân tích của [7], văn bản trong bộ dữ liệu SignboardText có hình dạng đa dạng, bao gồm các trường hợp văn bản nằm ngang (horizontal), văn bản có biên dạng tứ giác bất kỳ (arbitrary quadrilateral), cũng như văn bản đa hướng (multi-oriented). Đặc điểm này phản ánh sát với bối cảnh thực tế của biển hiệu ngoài trời, nơi các dòng chữ có thể được bố trí nghiêng, cong hoặc không song song với trục ảnh, qua đó đặt ra thách thức đáng kể cho các phương pháp phát hiện và nhận dạng văn bản tiên tiến nay.

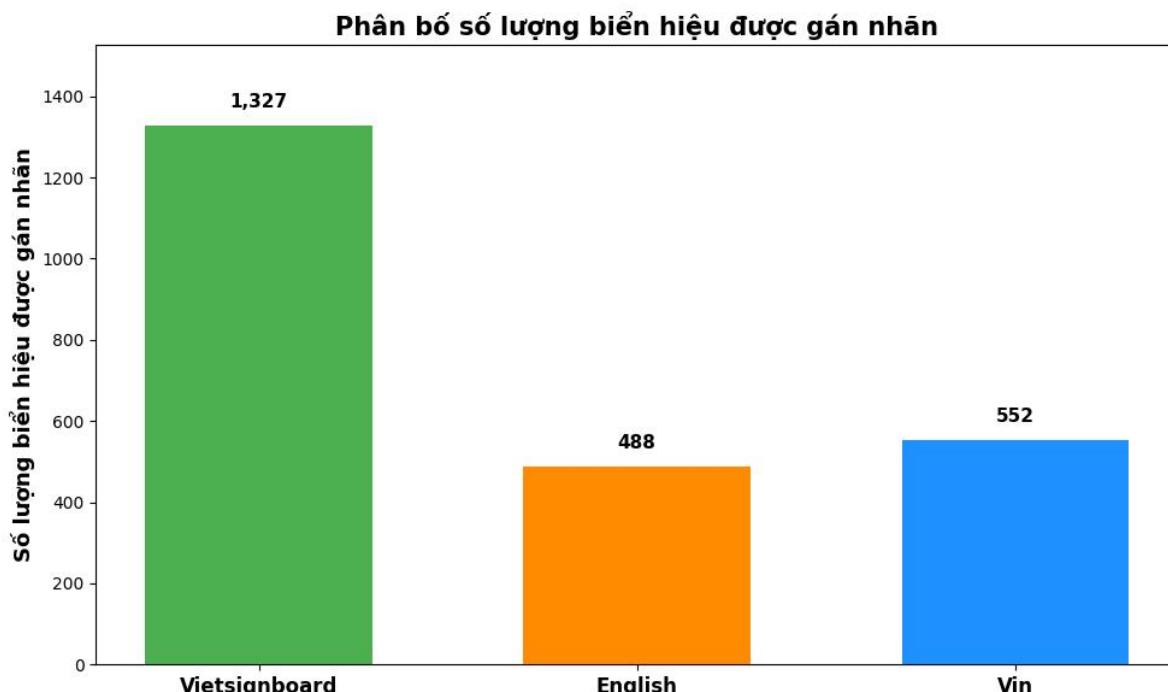
Mặc dù bộ dữ liệu SignboardText cung cấp hệ thống nhãn chi tiết cho văn bản ở nhiều cấp độ và hình dạng khác nhau, các nhãn (annotation) này vẫn chỉ tập trung vào các vùng văn bản, chưa bao quát toàn bộ đối tượng biển hiệu chứa văn bản. Trong khi đó, theo quy trình xử lý (pipeline) được đề xuất trong khóa luận, phát hiện biển hiệu



Hình 4.2: Phân bố số lượng thể hiện văn bản (text instances) theo cấp độ nhãn (word-level và line-level) trong các tập con của SignboardText [7]

đóng vai trò là bước tiền đề cho các giai đoạn phát hiện và nhận dạng văn bản phía sau. Do đó, nhằm hỗ trợ đánh giá giai đoạn phát hiện biển hiệu, khóa luận tiến hành mở rộng tập dữ liệu SignboardText bằng cách bổ sung lớp nhãn cho đối tượng biển hiệu. Cụ thể, toàn bộ 2.123 ảnh thuộc ba tập con Vietsignboard, English và Vin đã được gán nhãn thủ công, với sự hỗ trợ của công cụ PPOCRLLabel [11], để xác định các vùng chứa của biển hiệu trong ảnh. Trong đó, số lượng đối tượng biển hiệu được gán nhãn trong các tập Vietsignboard, English và Vin lần lượt là 1.327, 488 và 552. Phân bố số lượng các đối tượng này theo từng tập con được minh họa trong Hình 4.3.

Nhằm đánh giá mối quan hệ giữa các vùng văn bản và vùng biển hiệu trong tập dữ liệu SignboardText, khóa luận tiến hành thống kê số lượng và tỷ lệ phần trăm văn bản nằm trong vùng biển hiệu so với toàn bộ văn bản xuất hiện trong ảnh trên từng tập con của bộ dữ liệu. Dựa trên kết quả thống kê được trình bày trong Bảng 4.1, có thể nhận thấy rằng phần lớn văn bản trong tập dữ liệu SignboardText nằm bên trong các vùng biển hiệu đã được gán nhãn. Cụ thể, trên toàn bộ tập dữ liệu, SignboardText bao gồm



Hình 4.3: Phân bố số lượng đối tượng biển hiệu theo từng tập con của SignboardText [7]

68 899 văn bản ở cấp độ từ (word-level) và 10 950 văn bản ở cấp độ dòng (line-level). Trong số đó, 43 297 từ và 6 303 dòng văn bản nằm trong các vùng biển hiệu, tương ứng với tỷ lệ 62,84% ở mức từ và 57,56% ở mức dòng. Kết quả này cho thấy tập dữ liệu SignboardText phù hợp để đánh giá quy trình xử lý đầu-cuối (pipeline end-to-end) phát hiện và nhận dạng văn bản trên biển hiệu được đề xuất trong khóa luận.

Trong khuôn khổ khóa luận này, việc mở rộng tập dữ liệu chủ yếu tập trung vào bổ sung nhãn đối tượng biển hiệu (signboard annotation) cho tập dữ liệu ảnh tĩnh SignboardText hiện có, nhằm phục vụ trực tiếp cho quá trình huấn luyện và đánh giá mô hình. Bên cạnh đó, một tập dữ liệu video được thu thập trong môi trường đường phố Việt Nam, được sử dụng với mục đích minh họa cũng như kiểm tra khả năng tổng quát hóa của quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất trong bối cảnh thực tế.

Bảng 4.1: Thống kê số lượng và tỷ lệ văn bản nằm trong vùng biển hiệu so với toàn bộ văn bản trong ảnh trên các tập con của SignboardText

	Vietsignboard		English		Vin		Tổng	
	word	line	word	line	word	line	word	line
Số lượng văn bản trong ảnh	48638	10950	3646	-	16615	-	68899	10950
Số lượng văn bản trên biển hiệu	31374	6303	2613	-	9310	-	43297	6303
Tỷ lệ (%)	64.50	57.56	71.68	-	56.03	-	62.84	57.56

4.2 Thiết lập thực nghiệm

Trong khóa luận này, quá trình thực nghiệm được tổ chức theo hướng đánh giá từng giai đoạn một cách độc lập trong pipeline phát hiện và nhận dạng văn bản trên biển hiệu. Cách tổ chức này nhằm cho phép đánh giá độc lập hiệu quả của từng thành phần, đồng thời giảm chi phí huấn luyện và yêu cầu tài nguyên tính toán khi phải làm việc với nhiều mô hình khác nhau. Cụ thể, các thí nghiệm được thiết kế để lần lượt khảo sát từng giai đoạn chính, từ phát hiện biển hiệu, phát hiện và nhận dạng văn bản trên biển hiệu, cho đến việc xây dựng quy trình xử lý đầu-cuối (pipeline end-to-end). Kết quả thực nghiệm ở mỗi giai đoạn được sử dụng làm cơ sở để lựa chọn mô hình phù hợp, từ đó kết hợp và hình thành pipeline hoàn chỉnh cho bài toán đặt ra. Việc lựa chọn mô hình được thực hiện dựa trên sự cân bằng giữa độ chính xác, tốc độ xử lý và độ phức tạp mô hình, nhằm đảm bảo tính khả thi khi áp dụng pipeline trong bối cảnh xử lý video đường phố thực tế. Trên cơ sở này, các thiết lập thực nghiệm chi tiết cho từng giai đoạn sẽ được trình bày trong các mục tiếp theo.

4.2.1 Phát hiện biển hiệu

Trong quy trình xử lý (pipeline) phát hiện và nhận dạng văn bản trên biển hiệu, phát hiện biển hiệu đóng vai trò là bước khởi đầu, có ảnh hưởng trực tiếp đến hiệu quả của các giai đoạn xử lý phía sau. Do đặc thù về bối cảnh thu thập dữ liệu và sự khác biệt về miền dữ liệu so với các tập dữ liệu phát hiện đối tượng phổ biến, khóa luận tiến hành tinh chỉnh (fine-tuning) toàn bộ các mô hình khảo sát ở giai đoạn này nhằm đảm bảo

khả năng thích ứng với môi trường đường phố Việt Nam.

Các mô hình phát hiện biển hiệu được phân nhóm dựa trên dạng biểu diễn đầu ra, bao gồm: (i) các phương pháp dự đoán vùng bao chữ nhật (rectangle bounding box), (ii) các phương pháp dự đoán vùng bao định hướng (oriented bounding box - OBB), và (iii) các phương pháp dựa trên phân đoạn ngữ nghĩa để xác định vùng biển hiệu dưới dạng đa giác (polygon). Việc phân nhóm này cho phép đánh giá một cách có hệ thống các đặc điểm và ưu điểm của từng hướng tiếp cận trong bối cảnh bài toán đặt ra. Song song với đó, trong mỗi nhóm mô hình, các phương pháp được so sánh nhằm lựa chọn mô hình tốt nhất cho từng dạng đầu ra. Quá trình này tập trung đánh giá khả năng phát hiện trong điều kiện dữ liệu thực tế, đồng thời xem xét mức độ hiệu quả khi triển khai, làm cơ sở cho việc tích hợp các mô hình này vào quy trình xử lý (pipeline) và phục vụ các bước so sánh tổng thể ở các giai đoạn tiếp theo.

Bên cạnh đó, khóa luận tiến hành thực nghiệm bổ sung bước căn chỉnh biển hiệu (signboard alignment) đối với các mô hình có đầu ra là vùng bao định hướng (Oriented Bounding Box - OBB) hoặc đa giác (polygon). Trong thiết lập này, vùng biển hiệu sau khi được phát hiện sẽ được biến đổi phối cảnh (perspective transformation) để đưa về dạng chuẩn, qua đó cho phép so sánh hiệu quả giữa trường hợp không căn chỉnh và có căn chỉnh biển hiệu. Thiết lập này nhằm đánh giá mức độ ảnh hưởng của bước căn chỉnh đối với chất lượng dữ liệu đầu vào cho các giai đoạn phát hiện và nhận dạng văn bản phía sau. Hình 4.4 minh họa ví dụ quá trình căn chỉnh biển hiệu, trong đó vùng biển hiệu được phát hiện với đầu ra dạng đa giác (polygon) được biến đổi phối cảnh để đưa về dạng hình chữ nhật chuẩn, phục vụ cho các bước phát hiện và nhận dạng văn bản tiếp theo.

4.2.2 Phát hiện và nhận dạng văn bản trên biển hiệu

Bài toán phát hiện và nhận dạng văn bản trên biển hiệu thực hiện hai nhiệm vụ chính: xác định vị trí các vùng văn bản và nhận dạng nội dung bên trong. Để giải quyết bài toán này, khóa luận khảo sát theo hai hướng tiếp cận: hai giai đoạn (Two-Stage) và một giai đoạn (One-Stage). Các mô hình tiền huấn luyện (pretrained model) được sử dụng



Hình 4.4: Hình ảnh minh họa quá trình căn chỉnh biển hiệu (signboard alignment)

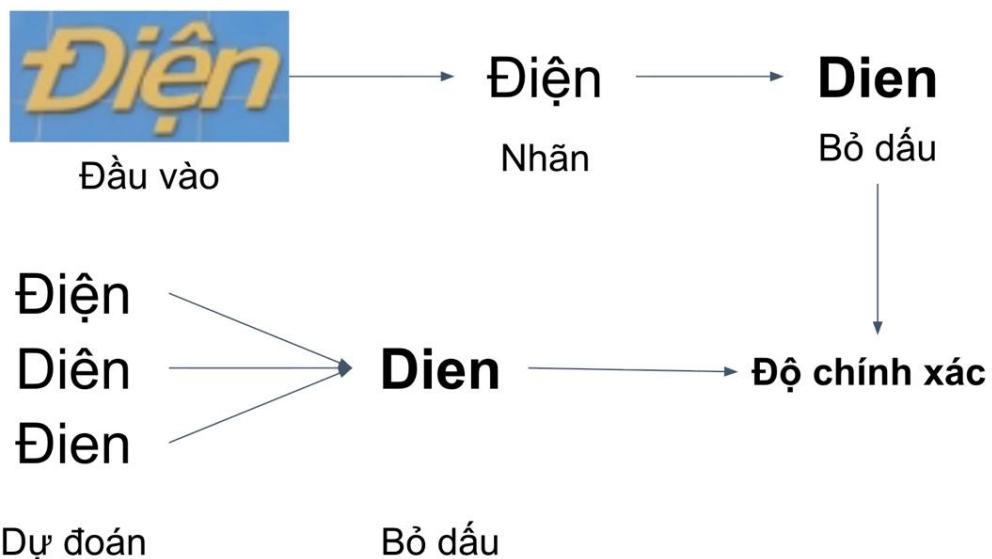
làm cơ sở đánh giá, từ đó lựa chọn hướng tiếp cận phù hợp nhằm tiết kiệm thời gian tinh chỉnh (fine-tune) mô hình. Bên cạnh đó, ở giai đoạn này, khóa luận thực nghiệm với tất cả các văn bản xuất hiện trong ảnh, thay vì chỉ giới hạn ở văn bản trên biển hiệu trong tập SignboardText mở rộng. Điều này giúp mở rộng dữ liệu đánh giá, từ đó cải thiện độ tin cậy của kết quả thực nghiệm.

4.2.2.1 Hướng tiếp cận hai giai đoạn (Two-Stage)

Phát hiện văn bản Trong bối cảnh bài toán phát hiện văn bản, khóa luận tiến hành thực nghiệm trên năm mô hình tiền huấn luyện, gồm PANet, DBNet++, TextPMs, FAST và KPN, được giới thiệu tại Mục 3.3.1. Dựa trên kết quả đánh giá hiệu suất các mô hình tiền huấn luyện (pretrained models), mô hình có hiệu năng cao nhất được lựa chọn để tiến hành tinh chỉnh (fine-tune) trên tập dữ liệu SignboardText, nhằm tối ưu hóa chi phí huấn luyện và đảm bảo tính khả thi trong quá trình thực nghiệm.

Bên cạnh các mô hình chuyên biệt cho phát hiện văn bản trong ảnh ngoại cảnh, hai phiên bản YOLOv8-OBB và YOLOv11-OBB, nổi bật nhờ sự cân bằng giữa độ chính xác và tốc độ xử lý, được áp dụng trong quá trình thực nghiệm. Do được huấn luyện ban đầu trên dữ liệu tổng quát (general object), các mô hình này cần được tinh chỉnh trực tiếp trên SignboardText để thích ứng với đặc thù văn bản trên biển hiệu, bao gồm các dòng chữ nghiêng và kích thước nhỏ. Sau quá trình tinh chỉnh, các mô hình được đánh giá tổng quan để lựa chọn ra mô hình tối ưu nhất cho việc tích hợp vào quy trình xử lý đầu cuối (pipeline end-to-end) đề xuất.

Đánh giá độ chính xác khi không xét dấu



Hình 4.5: Hình ảnh minh họa quá trình loại bỏ dấu tiếng Việt trên dữ liệu nhãn và kết quả đầu ra của mô hình

Nhận dạng nội dung văn bản Quá trình thực nghiệm cho bài toán nhận dạng được thực hiện trên các vùng văn bản đã được cắt ra từ tập SignboardText. Dựa trên chiến lược áp dụng cho bài toán phát hiện văn bản, các mô hình tiền huấn luyện (pretrained models) được sử dụng làm cơ sở đánh giá. Tuy nhiên, do các mô hình này được huấn luyện trên dữ liệu tiếng Anh, khóa luận thực hiện tiền xử lý nhằm phù hợp với văn bản tiếng Việt bằng cách loại bỏ dấu tiếng Việt và các ký tự đặc biệt trong dữ liệu nhãn cũng như kết quả đầu ra của mô hình. Hình 4.5 minh họa quá trình loại bỏ dấu tiếng Việt. Trên cơ sở kết quả đánh giá, ba mô hình có hiệu năng tốt nhất được lựa chọn để tiến hành tinh chỉnh trên tập SignboardText. Việc lựa chọn ba mô hình nhằm đánh giá khả năng tổng quát hóa trên văn bản tiếng Việt, đặc biệt khi số lượng ký tự tăng lên.

4.2.2.2 Hướng tiếp cận một giai đoạn (One-Stage)

Sau khi xác định mô hình tiền huấn luyện (pretrained model) tối ưu cho nhiệm vụ phát hiện và nhận dạng văn bản trong hướng tiếp cận hai giai đoạn, hai mô hình này được tích hợp để hình thành quy trình xử lý đầu-cuối (pipeline end-to-end), nhằm so

sánh với các mô hình tiền huấn luyện (pretrained model) một giai đoạn (one-stage) được trình bày tại Mục 3.3.3. Việc đánh giá này cho phép xác định hiệu quả của pipeline đầu-cuối (end-to-end) so với các mô hình một giai đoạn (one-stage), đồng thời cung cấp cơ sở để lựa chọn hướng tiếp cận phù hợp cho quá trình tinh chỉnh (fine-tune) trên tập SignboardText. Kết quả so sánh cũng đem lại cái nhìn tổng quan về sự khác biệt hiệu năng giữa chiến lược two-stage và one-stage, cũng như tối ưu hóa chi phí huấn luyện.

4.2.3 Phân chia bộ dữ liệu cho tập thực nghiệm

Nhằm đánh giá hiệu năng của các mô hình tiền huấn luyện cho các bài toán phát hiện văn bản, nhận dạng văn bản và phát hiện-nhận dạng văn bản đầu-cuối (end-to-end), toàn bộ dữ liệu của ba tập con Vietsignboard, English và Vin thuộc tập dữ liệu SignboardText mở rộng được sử dụng cho quá trình thực nghiệm. Việc sử dụng đầy đủ dữ liệu nhằm đảm bảo tính tin cậy và khả năng tổng quát hóa của các mô hình tiền huấn luyện trên dữ liệu biển hiệu đa ngôn ngữ.

Đối với giai đoạn tinh chỉnh mô hình, mỗi tập con Vietsignboard, English và Vin của tập dữ liệu SignboardText mở rộng được phân chia độc lập thành ba tập phục vụ cho huấn luyện, kiểm định và kiểm tra. Cụ thể, mỗi tập con được chia theo cùng một tỷ lệ gồm 65% cho tập huấn luyện, 15% cho tập kiểm định và 20% cho tập kiểm tra.

Sau khi hoàn tất việc phân chia dữ liệu hình ảnh cho các bài toán phát hiện biển hiệu và phát hiện văn bản, các vùng văn bản tương ứng tiếp tục được cắt từ các tập huấn luyện, kiểm định và kiểm tra để phục vụ cho bài toán nhận dạng văn bản. Số lượng cụ thể của từng tập được trình bày như sau:

- **Tập huấn luyện (Train set):** Chiếm 65% dữ liệu của mỗi tập con, với tổng cộng 1357 hình ảnh, trong đó Vietsignboard, English và Vin lần lượt gồm 764, 263 và 329 hình ảnh. Từ các hình ảnh này, các vùng văn bản được cắt ra với số lượng tương ứng là 31480, 2265 và 10493. Tập huấn luyện được sử dụng cho quá trình học tham số của mô hình và yêu cầu đảm bảo đủ số lượng dữ liệu để mô hình hội tụ ổn định.

- **Tập kiểm định (Validation set):** Chiếm 15% dữ liệu của mỗi tập con, bao gồm tổng cộng 340 hình ảnh, trong đó Vietsignboard, English và Vin lần lượt có 191, 66 và 83 hình ảnh. Số lượng vùng văn bản được trích xuất tương ứng là 7427, 610 và 2624. Tập này được sử dụng để tinh chỉnh siêu tham số và theo dõi hiệu năng mô hình trong quá trình huấn luyện.
- **Tập kiểm tra (Test set):** Chiếm 20% dữ liệu của mỗi tập con, với tổng cộng 340 hình ảnh, trong đó Vietsignboard, English và Vin lần lượt gồm 239, 82 và 104 hình ảnh. Từ các hình ảnh này, các vùng văn bản được cắt ra với số lượng tương ứng là 9730, 767 và 3491. Tập kiểm tra được sử dụng để đánh giá hiệu năng của mô hình sau khi hoàn tất quá trình tinh chỉnh.

Việc phân chia dữ liệu theo từng tập con Vietsignboard, English và Vin một cách độc lập nhằm đảm bảo mỗi tập con đều được phân bổ nhất quán vào các tập huấn luyện, kiểm định và kiểm tra, qua đó tránh hiện tượng phân bố lệch dữ liệu giữa các tập thực nghiệm. Nhờ vậy, hiệu năng mô hình có thể được đánh giá một cách tin cậy và rõ ràng trên dữ liệu tiếng Việt và tiếng Anh.

4.3 Tiền xử lý dữ liệu

Trong giai đoạn tinh chỉnh mô hình, với tính chất khảo sát và so sánh các mô hình hiện đại cho các tác vụ phát hiện văn bản và nhận dạng nội dung văn bản, mỗi mô hình được huấn luyện và đánh giá theo đúng quy trình tiền xử lý được đề xuất trong công trình gốc.

Trong khi đó, bài toán phát hiện biến hiệu vẫn áp dụng nhất quán các bước tiền xử lý ảnh như chuẩn hóa kích thước và giá trị đầu vào theo cấu hình của từng mô hình gốc. Đối với các phép tăng cường dữ liệu, khóa luận lựa chọn và áp dụng khác biệt giữa các nhóm kiến trúc phương pháp. Cụ thể, một số mô hình như DETR và RT-DETRv2 áp dụng các phép tăng cường liên quan đến biến đổi hình học và phối cảnh nhằm cải thiện khả năng phát hiện đối tượng trong các điều kiện chụp khác nhau. Các mô hình phân đoạn như SegFormer và Mask2Former ưu tiên các phép tăng cường liên quan đến

biến dạng, hướng và điều kiện chiếu sáng nhằm nâng cao khả năng khái quát hóa ở mức pixel. Đối với các phiên bản YOLO [37], quy trình tiền xử lý và tăng cường dữ liệu được áp dụng theo thiết lập được đề xuất trong công trình gốc.

4.4 Độ đo đánh giá

Để đánh giá chính xác hiệu quả và độ tin cậy của hệ thống phát hiện và nhận dạng văn bản trên biển hiệu, khóa luận sử dụng các độ đo phổ biến, phù hợp với từng giai đoạn xử lý. Chất lượng phát hiện biển hiệu được đánh giá thông qua mean Average Precision (mAP) cho các hộp giới hạn (bounding box) và mean Intersection over Union (mIoU) cho các vùng phân đoạn ngữ nghĩa của đối tượng biển hiệu. Đối với bài toán phát hiện văn bản, các độ đo Precision, Recall và Hmean theo giao thức **TedEval** [17] phản ánh khả năng xác định chính xác vị trí các vùng văn bản. Tỷ lệ nhận dạng chính xác chuỗi ký tự được đánh giá bằng Accuracy. Đặc biệt, nhằm đánh giá toàn diện hiệu quả hệ thống, bài toán phát hiện và nhận dạng văn bản đầu-cuối (end-to-end) sử dụng độ đo Hmean, kết hợp cả tiêu chí về vị trí (IoU) và nhận dạng nội dung, nhằm phản ánh đầy đủ chất lượng hệ thống.

4.4.1 Phát hiện biển hiệu

Mean Average Precision (mAP) Chỉ số mAP đánh giá khả năng phát hiện chính xác các vùng biển hiệu dựa trên đường cong Precision–Recall. Cụ thể, Average Precision (AP) cho một lớp đối tượng được tính bằng diện tích dưới đường cong này, sử dụng phương pháp nội suy toàn điểm (all-point interpolation).

$$AP = \sum_n (r_{n+1} - r_n) \cdot \max_{\tilde{r} \geq r_{n+1}} p(\tilde{r}) \quad (4.1)$$

Trong đó, $p(\tilde{r})$ và \tilde{r} lần lượt là giá trị precision và recall tại một ngưỡng nhất định. Giá trị mAP cuối cùng được tính bằng trung bình của AP trên tất cả các lớp. Trong khóa luận này, các ngưỡng IoU áp dụng bao gồm 0.5 (mAP@0.5) và trung bình từ 0.5 đến 0.95 (mAP@[0.5:0.95]), được tính toán thông qua công cụ [29].

Mean Intersection over Union (mIoU) Nhằm đo lường mức độ trùng khớp giữa vùng dự đoán và vùng thực tế (ground truth) cho đối tượng biển hiệu, khóa luận sử dụng chỉ số mIoU, được tính bằng trung bình các IoU của từng lớp, ở cấp độ pixel.

$$\text{IoU} = \frac{TP}{TP + FP + FN} \quad (4.2)$$

Trong đó: Trong đó:

- TP (True Positive): Số pixel được dự đoán đúng là biển hiệu
- FP (False Positive): Số pixel bị dự đoán nhầm là biển hiệu (thực tế là nền hoặc đối tượng khác)
- FN (False Negative): Số pixel thuộc biển hiệu thực tế nhưng bị dự đoán sai (thành nền hoặc đối tượng khác)

4.4.2 Phát hiện văn bản

Dựa trên nghiên cứu TedEval [17], phương pháp đánh giá này được xem là phù hợp và ổn định trong việc đánh giá chất lượng phát hiện văn bản trong ảnh ngoại cảnh. Do đó, khóa luận sử dụng các độ đo Precision (P), Recall (R) và Hmean (F1-score) cho việc đánh giá.

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (4.3)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (4.4)$$

$$\text{Hmean} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4.5)$$

Trong đó, TP, FP và FN lần lượt là True Positive, False Positive và False Negative.

4.4.3 Nhận dạng văn bản

Nhằm phản ánh hiệu quả nhận dạng văn bản trong ảnh ngoại cảnh, khóa luận sử dụng độ đo Accuracy, đánh giá tỷ lệ chuỗi ký tự dự đoán chính xác.

$$\text{Accuracy} = \frac{\text{Số từ (hoặc dòng) nhận dạng đúng}}{\text{Tổng số từ (hoặc dòng)}} \quad (4.6)$$

Trong đó, một từ hoặc dòng chỉ được tính là đúng khi chuỗi ký tự dự đoán khớp hoàn toàn với chuỗi ký tự trong thực tế (ground truth).

4.4.4 Phát hiện và nhận dạng văn bản đầu-cuối (end-to-end)

Để đánh giá toàn diện khả năng phát hiện và nhận dạng, khóa luận sử dụng độ đo Hmean đầu-cuối (end-to-end), với cặp giá trị dự đoán và thực tế chỉ được xét khi thỏa hai điều kiện, gồm vùng phát hiện trùng với nhãn (ground truth) theo một ngưỡng IoU nhất định (ví dụ: $\text{IoU} \geq 0.5$), và chuỗi ký tự nhận dạng khớp với nhãn.

$$H\text{mean}_{e2e} = \frac{2 \cdot \text{Precision}_{e2e} \cdot \text{Recall}_{e2e}}{\text{Precision}_{e2e} + \text{Recall}_{e2e}} \quad (4.7)$$

Trong đó, Precision_{e2e} và Recall_{e2e} được xác định dựa trên các thể hiện (instances) thỏa mãn điều kiện, nhằm phản ánh đồng thời chất lượng phát hiện và nhận dạng của hệ thống.

4.5 Kết quả thực nghiệm

Trong phần này, khóa luận trình bày chi tiết kết quả thực nghiệm dựa trên các thiết lập được mô tả tại Mục 4.2. Trước tiên, các kết quả được đánh giá độc lập theo từng giai đoạn. Với giai đoạn phát hiện biển hiệu, mô hình tốt nhất trong mỗi nhóm đầu ra được lựa chọn, sau đó các mô hình tối ưu cho từng giai đoạn khác cũng được xác định. Các mô hình này được tích hợp vào pipeline và đánh giá tổng thể hiệu quả hệ thống phát hiện và nhận dạng văn bản trên biển hiệu.

Bảng 4.2: Hiệu suất phát hiện biển hiệu của các mô hình với đầu ra dạng hình chữ nhật (rectangle bounding box). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Year	Params(M)	FPS	mAP ₅₀ (%)	mAP ₅₀₋₉₅ (%)
DETR	2020	41.50	42.82	<u>89.30</u>	68.95
YOLOv8	2023	3.01	133.01	86.85	<u>74.20</u>
RT-DETRv2	2024	42.73	26.44	90.71	81.22
YOLOv11	2024	2.59	<u>96.06</u>	88.00	73.47

4.5.1 Kết quả mô hình phát hiện biển hiệu đã tinh chỉnh (fine-tuning)

Như đã đề cập tại Mục 4.2.1, tất cả các mô hình khảo sát trong giai đoạn phát hiện biển hiệu được tinh chỉnh (fine-tune) trên tập dữ liệu SignboardText đã được mở rộng, và các mô hình được phân loại thành ba nhóm dựa trên dạng biểu diễn đầu ra. Việc phân nhóm này cho phép đánh giá một cách hệ thống các đặc điểm và ưu điểm của từng hướng tiếp cận, đồng thời so sánh hiệu năng giữa các mô hình trong cùng nhóm mà không gây nhầm lẫn giữa các dạng đầu ra khác nhau.

Bảng 4.2 trình bày kết quả các mô hình phát hiện biển hiệu với đầu ra dạng hình chữ nhật (rectangle bounding box). Trong nhóm này, RT-DETRv2 đạt hiệu suất tốt nhất với các chỉ số mAP₅₀ và mAP₅₀₋₉₅ lần lượt là **90.71%** và **81.22%**, cho thấy khả năng phát hiện chính xác các biển hiệu có kích thước đa dạng. Đồng thời, chỉ số FPS của RT-DETRv2 cho phép mô hình thực hiện trong thời gian thực, cho thấy mô hình này phù hợp để tích hợp vào pipeline đề xuất cho giai đoạn phát hiện biển hiệu. Bên cạnh đó, các mô hình còn lại như DETR, YOLOv8 và YOLOv11 thể hiện sự cân bằng giữa độ chính xác và tốc độ xử lý, trong đó YOLOv8 nổi bật với tốc độ **133.01** FPS, phù hợp cho các ứng dụng yêu cầu xử lý thời gian thực. Nhìn chung, các kết quả này cung cấp cơ sở để lựa chọn mô hình tối ưu, trong đó RT-DETRv2 được xem là lựa chọn ưu tiên cho đầu ra dạng hình chữ nhật, về cả độ chính xác lẫn hiệu năng thực thi.

Bảng 4.3 trình bày kết quả các mô hình dự đoán vùng bao định hướng. Cả hai mô hình YOLOv8-OBB và YOLOv11-OBB đều đạt giá trị mAP₅₀ và mAP₅₀₋₉₅ cao. Trong đó, YOLOv11-OBB nổi bật với mAP₅₀₋₉₅ **80.78%** và là lựa chọn tối ưu để tích hợp vào

Bảng 4.3: Hiệu suất phát hiện biển hiệu của các mô hình với đầu ra dạng vùng bao định hướng (Oriented Bounding Box - OBB). Chỉ số tốt nhất được đánh dấu đậm

Model	Year	Params(M)	FPS	mAP ₅₀ (%)	mAP ₅₀₋₉₅ (%)
YOLOv8-OBB	2023	3.01	133.01	93.04	79.08
YOLOv11-OBB	2024	2.59	96.06	93.20	80.78

Bảng 4.4: Hiệu suất phát hiện vùng biển hiệu của các mô hình phân đoạn ngữ nghĩa. Chỉ số tốt nhất được đánh dấu đậm.

Model	Year	Params(M)	FPS	mIOU(%)	mAccuracy(%)
SegFormer	2021	3.8	97.5	89.03	93.92
Mask2Former	2022	47.4	15.9	90.48	94.44

pipeline, nhờ khả năng phát hiện chính xác các biển hiệu nghiêng hoặc không vuông góc với camera. Kết quả này đồng thời cải thiện chất lượng dữ liệu đầu vào cho giai đoạn phát hiện và nhận dạng văn bản.

Bên cạnh các mô hình phát hiện dựa trên bounding box, khóa luận tiến hành thực nghiệm mô hình SegFormer và Mask2Former nhằm xác định chính xác vùng biển hiệu, giúp giảm chi phí tính toán cho các giai đoạn phát hiện và nhận dạng văn bản, nhờ hạn chế phần nền thừa không liên quan, đồng thời hữu ích với các biển hiệu nghiêng hoặc có hình dạng không đều. Theo kết quả trong Bảng 4.4, Mask2Former thể hiện ưu thế rõ rệt về độ chính xác khi đạt các chỉ số mIoU và mAccuracy cao hơn SegFormer, lần lượt là **90.48%** và **94.44%**. Mặt khác, SegFormer lại nổi bật với tốc độ xử lý cao hơn đáng kể với chỉ số FPS **97.5**, chứng tỏ sự cân bằng giữa độ chính xác và hiệu năng thực thi, phù hợp cho các ứng dụng yêu cầu thời gian thực. Do đó, SegFormer được lựa chọn làm mô hình tối ưu cho giai đoạn phân đoạn biển hiệu trong quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất, đảm bảo vừa duy trì độ chính xác vừa đáp ứng yêu cầu tốc độ xử lý.

4.5.2 Kết quả mô hình tiền huấn luyện đối với bài toán phát hiện văn bản

Nhằm tối ưu chi phí huấn luyện và tài nguyên tính toán, thay vì tinh chỉnh toàn bộ các mô hình phát hiện văn bản tiên tiến hiện nay, khóa luận tiến hành đánh giá hiệu quả

của các mô hình tiền huấn luyện trên tập dữ liệu SignboardText đã được mở rộng. Cụ thể, các mô hình được kiểm tra trên ba tập con VietSignboard, English và Vin, với hai cấp độ đánh giá là cấp độ từ (word-level) và cấp độ dòng (line-level). Theo thiết lập đã trình bày tại Mục 4.2.2, khóa luận thực nghiệm trên toàn bộ các vùng văn bản xuất hiện trong ảnh, thay vì chỉ giới hạn trong phạm vi biển hiệu. Trên cơ sở đó, mô hình tiền huấn luyện có hiệu suất tốt nhất sẽ được lựa chọn để tiếp tục tinh chỉnh (fine-tuning) cho bài toán phát hiện văn bản.

Đánh giá ở cấp độ từ (word-level) Kết quả đánh giá các mô hình tiền huấn luyện ở cấp độ từ được trình bày trong Bảng 4.5. Nhìn chung, hầu hết các mô hình đều cho kết quả khả quan trên cả ba tập con. Đáng chú ý, các mô hình vẫn cho kết quả tốt trên hai tập VietSignboard và Vin, mặc dù không được huấn luyện trực tiếp trên dữ liệu tiếng Việt, cho thấy khả năng tổng quát hóa tương đối tốt.

Trong số các mô hình được khảo sát, TextPMs thể hiện hiệu suất vượt trội trên cả ba tập con VietSignboard, English và Vin, với chỉ số Hmean lần lượt đạt **87.48%**, **78.20%** và **86.95%**. Đặc biệt, xét theo chỉ số Precision, TextPMs đạt giá trị cao trên hai tập VietSignboard và Vin, lần lượt là **90.27%** và **93.41%**, cho thấy phần lớn các vùng mô hình phát hiện đều là văn bản thật, phản ánh khả năng lọc nhiễu và giảm thiểu dự đoán sai. Bên cạnh độ chính xác, TextPMs cũng đạt tốc độ xử lý cao nhất trong số các mô hình được khảo sát, với chỉ số FPS **20.75**, thể hiện sự cân bằng hợp lý giữa hiệu quả phát hiện và hiệu năng thực thi. Ngoài ra, FAST và KPN cũng cho thấy kết quả đáng chú ý, khi lần lượt đạt Recall cao nhất trên hai tập con VietSignboard với **86.32%** và English với **85.90%**, tuy nhiên hiệu suất tổng thể và tốc độ xử lý vẫn kém hơn so với TextPMs.

Đánh giá ở cấp độ dòng (line-level) Bên cạnh đánh giá ở cấp độ từ (word-level), khóa luận tiếp tục khảo sát hiệu quả của các mô hình tiền huấn luyện ở cấp độ dòng (line-level) trên tập con VietSignboard, với kết quả được trình bày trong Bảng 4.6. Kết quả cho thấy hiệu suất của hầu hết các mô hình đều giảm đáng kể so với đánh giá ở cấp độ từ, thể hiện qua chỉ số Hmean thấp. Từ đó có thể nhận thấy rằng phần lớn các mô hình tiền huấn luyện này được tối ưu chủ yếu cho bài toán phát hiện văn bản ở cấp độ từ, và

Bảng 4.5: Hiệu suất mô hình tiền huấn luyện trong bài toán phát hiện văn bản ở cấp độ từ (word-level) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText mở rộng. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Year	Params(M)	Vietsignboard			English			Vin			FPS
			P	R	H	P	R	H	P	R	H	
PANet	2020	12.25	81.00	82.25	81.62	61.00	72.56	66.28	81.71	75.82	78.66	11.58
DBNet++	2022	26.43	<u>89.86</u>	80.31	84.82	<u>73.38</u>	60.95	65.59	<u>91.52</u>	74.18	<u>81.94</u>	18.69
TextPMs	2022	36.43	90.27	<u>84.85</u>	87.48	78.82	77.58	78.20	93.41	81.32	86.95	20.75
FAST	2023	10.58	83.98	86.32	<u>85.13</u>	64.34	80.25	71.42	84.45	79.09	81.69	15.30
KPN	2023	58.24	81.19	81.85	81.52	63.49	85.90	<u>73.01</u>	83.49	78.37	80.85	4.17

Bảng 4.6: Hiệu suất mô hình tiền huấn luyện trong bài toán phát hiện văn bản ở cấp độ dòng (line-level) trên tập con VietSignboard của tập dữ liệu SignboardText mở rộng. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Vietsignboard		
	P	R	H
PANet	21.68	74.62	33.60
DBNet++	22.14	70.54	33.70
TextPMs	<u>22.84</u>	<u>75.27</u>	<u>35.04</u>
FAST	21.91	79.92	34.39
KPN	25.23	78.48	38.18

chưa phù hợp khi áp dụng trực tiếp cho trường hợp phát hiện ở cấp độ dòng mà không có điều chỉnh hoặc huấn luyện bổ sung.

Dựa trên kết quả đánh giá năm mô hình tiền huấn luyện trên ba tập con VietSignboard, English và Vin, có thể thấy TextPMs là mô hình đạt hiệu suất tổng thể tốt nhất, xét trên cả độ chính xác và tốc độ xử lý, ngay cả trong bối cảnh dữ liệu tiếng Việt chưa xuất hiện trong quá trình huấn luyện ban đầu. Bên cạnh đó, sự khác biệt rõ rệt giữa kết quả ở cấp độ từ và cấp độ dòng, cung cấp cơ sở thực nghiệm quan trọng để khóa luận lựa chọn chiến lược tinh chỉnh các mô hình theo hướng cấp đồ từ (word-level) cho bài toán phát hiện văn bản trong quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất.

4.5.3 Kết quả mô hình tiền huấn luyện đối với bài toán nhận dạng văn bản

Kế thừa chiến lược đánh giá trong bài toán phát hiện văn bản, khóa luận tiếp tục khảo sát hiệu suất của các mô hình tiền huấn luyện trong bài toán nhận dạng văn bản. Cụ thể,

năm mô hình ViTSTR, PARSeq, CDistNet, SMTR và SVTRv2 được thực nghiệm trên các vùng văn bản đã được cắt ra từ ba tập con VietSignboard, English và Vin. Việc đánh giá được thực hiện ở hai cấp độ: cấp độ từ (word-level) và cấp độ dòng (line-level). Tuy nhiên, như đã trình bày trong thiết lập thực nghiệm tại Mục 4.2.2, do các mô hình nhận dạng được huấn luyện chủ yếu trên dữ liệu tiếng Anh, khóa luận tiến hành bước tiền xử lý nhằm phù hợp với văn bản tiếng Việt. Cụ thể, dấu tiếng Việt và các ký tự đặc biệt được loại bỏ trong cả dữ liệu nhãn và kết quả dự đoán của mô hình, nhằm đảm bảo tính nhất quán trong quá trình đánh giá.

Trên cơ sở đó, Bảng 4.7 trình bày kết quả hiệu suất của các mô hình tiền huấn luyện trên ba tập con ở cả hai cấp độ đánh giá. Nhìn chung, PARSeq, SMTR và SVTRv2 thể hiện hiệu suất vượt trội so với ViTSTR và CDistNet trên cả ba tập dữ liệu, cho thấy khả năng tổng quát hóa tốt hơn trong bối cảnh văn bản đa ngôn ngữ.

Đặc biệt ở cấp độ từ (word-level) SVTRv2 đạt độ chính xác cao nhất trên hai tập VietSignboard và English, lần lượt là **80.82%** và **78.33%**, trong khi PARSeq cho kết quả tốt hơn trên tập Vin với độ chính xác **72.96%**. Điều này cho thấy hai mô hình đều thể hiện khả năng nhận dạng từ đơn lẻ hiệu quả. Trong khi ở cấp độ dòng (line-level), SVTRv2 và SMTR tiếp tục duy trì hiệu suất cao trên tập VietSignboard, lần lượt đạt **65.64%** và **64.54%** so với các mô hình còn lại. Kết quả này phản ánh hiệu quả của chiến lược điều chỉnh đầu vào đa kích thước (multi-size resizing) được áp dụng trong hai mô hình, như đã trình bày tại Mục 3.3.2. Bên cạnh độ chính xác, tốc độ suy luận cũng là một yếu tố quan trọng trong bối cảnh triển khai thực tế. Đáng chú ý, mặc dù ViTSTR có số lượng tham số lớn nhất, mô hình này đạt tốc độ suy luận nhanh nhất trong số các phương pháp được so sánh, phản ánh thiết kế kiến trúc tương đối đơn giản và hiệu quả.

Dựa trên thiết kế thực nghiệm đã trình bày, khóa luận lựa chọn ba mô hình PARSeq, SMTR và SVTRv2 để tiến hành tinh chỉnh trên tập dữ liệu SignboardText đã được mở rộng. Đồng thời, do các mô hình thể hiện sự hiệu quả rõ rệt ở cấp độ từ, quá trình tinh chỉnh trong bài toán nhận dạng văn bản được thực hiện theo chiến lược word-level.

Bảng 4.7: Hiệu suất các mô hình tiền huấn luyện trong bài toán nhận dạng văn bản trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText đã được mở rộng, với hai cấp độ đánh giá từ (word-level) và dòng (line-level). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Year	Params(M)	Accuracy(%)						Speed(ms)	
			Vietsignboard		English		Vin			
			Word	Line	Word	Line	Word	Line		
ViTSTR	2021	85.48	56.70	29.01	48.71	-	50.82	-	9.80	
PARSeq	2022	23.83	<u>80.76</u>	59.08	<u>78.28</u>	-	72.96	-	<u>12.71</u>	
CDistNet	2024	65.46	63.25	42.43	68.13	-	56.39	-	120.33	
SMTR	2025	15.82	79.58	<u>64.54</u>	76.77	-	70.64	-	23.47	
SVTRv2	2025	21.02	80.82	65.64	78.33	-	72.34	-	18.81	

4.5.4 Kết quả so sánh hướng tiếp cận hai giai đoạn (two-stage) và một giai đoạn (one-stage) trong bài toán phát hiện và nhận dạng văn bản

Sau khi đánh giá riêng biệt các mô hình tiền huấn luyện cho hai nhiệm vụ phát hiện và nhận dạng văn bản trong hướng tiếp cận hai giai đoạn (two-stage), khóa luận tiến hành kết hợp hai mô hình có hiệu năng tốt nhất là TextPMs cho giai đoạn phát hiện và SVTRv2 cho giai đoạn nhận dạng nhằm xây dựng pipeline đầu-cuối (end-to-end). Pipeline này được sử dụng để so sánh với các mô hình tiền huấn luyện một giai đoạn (one-stage) tiêu biểu, bao gồm TESTR, DeepSolo, UNITS và DNTextSpotter. Mục tiêu của phép so sánh này là đánh giá hiệu quả tương đối giữa hai chiến lược tiếp cận, từ đó lựa chọn hướng tiếp cận phù hợp cho quá trình tinh chỉnh (fine-tune) trên tập dữ liệu SignboardText đã được mở rộng. Thiết lập thực nghiệm với hướng tiếp cận one-stage được trình bày tại Mục 4.2.2.2.

Bảng 4.8 trình bày kết quả so sánh giữa pipeline hai giai đoạn và các mô hình tiền huấn luyện một giai đoạn trên ba tập con VietSignboard, English và Vin, theo hai cấp độ đánh giá từ (word-level) và dòng (line-level).

Xét ở cấp độ từ, pipeline kết hợp TextPMs và SVTRv2 thể hiện ưu thế rõ rệt trên hai tập VietSignboard và Vin, với chỉ số Hmean lần lượt đạt **66.48%** và **68.50%**. Kết quả này cho thấy pipeline hai giai đoạn có khả năng tổng quát hóa tốt hơn trên văn bản tiếng Việt so với các mô hình one-stage còn lại. Trong khi đó, trên tập English, các mô hình

Bảng 4.8: So sánh hiệu suất mô hình tiền huấn luyện theo hai hướng tiếp cận hai giai đoạn (two-stage) và một giai đoạn (one-stage) trong bài toán phát hiện và nhận dạng văn bản trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText, với hai cấp độ đánh giá từ (word-level) và dòng (line-level). Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới

Model	Year	Params(M)	Hmean _{e2e} (%)						FPS	
			Vietsignboard		English		Vin			
			Word	Line	Word	Line	Word	Line		
TESTR	2022	49.48	48.77	7.43	60.43	-	50.11	-	11.11	
DeepSolo	2023	42.59	47.61	8.02	70.71	-	53.32	-	12.21	
UNITS	2023	101.00	63.19	<u>9.75</u>	88.00	-	64.88	-	1.28	
DNTTextSpotter	2025	42.73	49.17	9.20	<u>73.20</u>	-	53.57	-	12.40	
TextPMs + SVTRv2	-	57.45	<u>66.48</u>	9.79	68.59	-	68.50	-	10.21	

one-stage đạt kết quả cao hơn pipeline hai giai đoạn. Trong đó, UNITS cho hiệu suất nổi bật trên cả ba tập con. Tuy nhiên, mô hình này có số lượng tham số lớn với 101M và tốc độ xử lý thấp với 1.28 FPS, dẫn đến hạn chế đáng kể về khả năng triển khai trong các trường hợp ứng dụng yêu cầu hiệu năng thời gian hoặc tài nguyên tính toán hạn chế. Các mô hình one-stage còn lại như TESTR, DeepSolo và DNTTextSpotter có tốc độ xử lý cao hơn pipeline hai giai đoạn, tuy nhiên hiệu suất tổng thể (Hmean) trên các tập dữ liệu tiếng Việt thấp hơn đáng kể.

Tương tự xu hướng đã quan sát trong các thí nghiệm trước đó (Mục 4.5.2 và Mục 4.5.3), các mô hình one-stage nhìn chung không đạt hiệu quả cao khi đánh giá ở cấp độ dòng (line-level). Thậm chí, dù SVTRv2 cho kết quả tương đối tốt ở cấp độ dòng trên tập VietSignboard, pipeline hai giai đoạn vẫn chịu hạn chế cố hữu khi độ chính xác của giai đoạn nhận dạng phụ thuộc chặt chẽ vào chất lượng phát hiện văn bản ở giai đoạn trước.

Dựa trên kết quả phân tích và thực nghiệm, khóa luận lựa chọn hướng tiếp cận hai giai đoạn làm chiến lược chính cho pipeline phát hiện và nhận dạng văn bản trên biển hiệu. Mặc dù các mô hình one-stage đạt kết quả tốt hơn trên tập English, bài toán nghiên cứu hướng tới bối cảnh ứng dụng thực tế tại Việt Nam. Do đó, hiệu suất trên hai tập VietSignboard và Vin được ưu tiên trong quá trình lựa chọn hướng tiếp cận.

4.5.5 Kết quả mô hình phát hiện văn bản đã tinh chỉnh (fine-tuned)

Trên cơ sở kết quả đánh giá các mô hình tiền huấn luyện cho bài toán phát hiện văn bản được trình bày tại Mục 4.5.2, khóa luận tiến hành tinh chỉnh (fine-tune) ba mô hình TextPMs, YOLOv8-OBB và YOLOv11-OBB, được lựa chọn dựa trên sự cân bằng giữa độ chính xác, tốc độ xử lý và khả năng thích ứng với đặc thù văn bản trên biển hiệu, nhằm tối ưu hóa hiệu suất phát hiện văn bản trên tập dữ liệu SignboardText.

Bảng 4.9 trình bày hiệu suất của ba mô hình sau khi được tinh chỉnh trên các tập hình ảnh thuộc ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText mở rộng theo cấp độ từ (word-level). Kết quả cho thấy TextPMs tiếp tục khẳng định ưu thế, đạt chỉ số Hmean cao nhất trên cả ba tập dữ liệu lần lượt là **87.75%**, **83.09%** và **88.51%**. Đặc biệt, TextPMs cải thiện đáng kể chỉ số Recall sau khi tinh chỉnh, cho thấy khả năng phát hiện được nhiều vùng văn bản thực sự hơn.

Đáng chú ý, mặc dù YOLOv8 và YOLOv11 không phải là các kiến trúc chuyên biệt cho bài toán phát hiện văn bản trong ảnh ngoại cảnh, cả hai đều cho kết quả rất ấn tượng sau khi tinh chỉnh. YOLOv8-OBB thể hiện hiệu suất cạnh tranh trực tiếp với TextPMs, với Hmean đạt **87.39%**, **82.58%**, và **84.73%** trên ba tập, đồng thời vượt trội về chỉ số Precision trên tập Vietsignboard với **91.14%**. Kết quả này không chỉ phản ánh tính đa năng của kiến trúc YOLO mà còn cho thấy tiềm năng ứng dụng của chúng vào bài toán phát hiện văn bản khi được huấn luyện trên dữ liệu phù hợp.

Dựa trên kết quả tổng hợp, TextPMs tiếp tục là mô hình đạt hiệu suất cao nhất xét theo chỉ số Hmean, cho thấy tính hiệu quả và độ ổn định trên cả ba tập dữ liệu VietSignboard, English và Vin. Tuy nhiên, trong bối cảnh tích hợp vào một pipeline ứng dụng thực tế, việc lựa chọn mô hình không chỉ phụ thuộc vào độ chính xác mà còn cần xem xét đến tốc độ xử lý. Theo đó, YOLOv8-OBB nổi lên như một lựa chọn phù hợp khi vừa duy trì hiệu suất phát hiện tương đương với TextPMs (với chênh lệch Hmean không đáng kể trên các tập VietSignboard và English), vừa thể hiện ưu thế vượt trội về tốc độ xử lý. Cụ thể, kết quả đánh giá ở Bảng 4.3 cho thấy YOLOv8-OBB đạt tốc độ lên tới 133.01 FPS, cao hơn đáng kể so với TextPMs. Bên cạnh đó, sau khi tinh chỉnh, YOLOv8-OBB cũng cho kết quả tốt hơn YOLOv11-OBB trên cả ba tập dữ liệu. Do đó, YOLOv8-OBB

Bảng 4.9: Hiệu suất mô hình phát hiện văn bản đã tinh chỉnh. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Vietsignboard			English			Vin		
	P	R	H	P	R	H	P	R	H
TextPMs	90.36	85.29	87.75	83.80	82.39	83.09	92.98	84.46	88.51
YOLOv8-OBB	<u>91.14</u>	<u>83.94</u>	<u>87.39</u>	83.77	<u>81.42</u>	<u>82.58</u>	93.34	<u>77.57</u>	84.73
YOLOv11-OBB	91.49	82.97	87.02	<u>84.16</u>	78.70	81.34	<u>93.10</u>	76.84	84.19

là lựa chọn phù hợp để tích hợp vào quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất, khi đáp ứng được sự cân bằng tối ưu giữa độ chính xác cao và tốc độ xử lý nhanh.

4.5.6 Kết quả mô hình nhận dạng văn bản đã tinh chỉnh (fine-tuned)

Kết quả thực nghiệm đánh giá các mô hình tiền huấn luyện trong bài toán nhận dạng văn bản được trình bày tại Mục 4.5.3, khóa luận tiến hành tinh chỉnh ba mô hình PARSeq, SMTR và SVTRv2 trên các vùng văn bản được cắt ra từ tập dữ liệu SignboardText đã được mở rộng, nhằm nâng cao khả năng thích nghi với văn bản tiếng Việt trong bối cảnh biển hiệu đường phố Việt Nam và lựa chọn mô hình tối ưu cho giai đoạn nhận dạng văn bản trong pipeline đề xuất. Kết quả đánh giá hiệu suất của ba mô hình PARSeq, SMTR và SVTRv2 đã tinh chỉnh được trình bày trong Bảng 4.10, với hai tiêu chí đánh giá là Exact-match và Normalized-match. Exact-match đánh giá độ chính xác khi chuỗi dự đoán trùng khớp hoàn toàn với nhãn gốc. Trong khi đó, Normalized-match áp dụng cùng tiêu chí so sánh, nhưng thực hiện chuẩn hóa chữ hoa-chữ thường (lowercase) trước khi đánh giá, nhằm giảm ảnh hưởng của sự không nhất quán về chữ hoa-chữ thường, vốn thường xuất hiện trong văn bản trên biển hiệu thực tế.

Nhìn chung, cả ba mô hình PARSeq, SMTR và SVTRv2 đều đạt hiệu suất tương đối tốt trên ba tập con VietSignboard, English và Vin. Đặc biệt, kết quả trên hai tập VietSignboard và Vin cho thấy các mô hình có khả năng thích nghi và tổng quát hóa hiệu quả đối với văn bản tiếng Việt trong bối cảnh biển hiệu đường phố. Trong số đó, PARSeq thể hiện hiệu suất vượt trội khi đạt độ chính xác cao nhất ở cả hai tiêu chí Exact-match và Normalized-match, lần lượt là **79.19%** và **80.26%** trên tập VietSignboard, cũng như **76.95%** và **78.67%** trên tập Vin. Kết quả này phản ánh khả năng nhận dạng

Bảng 4.10: Hiệu suất mô hình nhận dạng văn bản đã tinh chỉnh. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model	Accuracy(%)					
	Vietsignboard		English		Vin	
	Exact-match	Norm-match	Exact-match	Norm-match	Exact-match	Norm-match
PARSeq	79.19	80.26	60.68	62.11	76.95	78.67
SMTR	<u>77.40</u>	<u>78.47</u>	<u>59.64</u>	<u>61.07</u>	75.06	76.58
SVTRv2	76.39	77.96	57.55	58.59	<u>76.46</u>	<u>78.10</u>

từ đơn lẻ hiệu quả và ổn định của PARSeq sau quá trình tinh chỉnh trên dữ liệu tiếng Việt.

Đáng chú ý, mặc dù SVTRv2 đạt hiệu suất cao ở giai đoạn tiền huấn luyện, mức độ cải thiện của mô hình sau khi tinh chỉnh lại thấp hơn so với PARSeq và SMTR. Điều này cho thấy khả năng thích nghi của SVTRv2 với miền dữ liệu biển hiệu tiếng Việt chưa thực sự vượt trội so với hai mô hình còn lại. Bên cạnh đó, kết quả trên tập English cho thấy hiệu suất của cả ba mô hình có xu hướng giảm nhẹ so với các tập tiếng Việt. Nguyên nhân có thể xuất phát từ việc các mô hình được tinh chỉnh chuyên sâu trên dữ liệu tiếng Việt với số lượng ký tự và biến thể phong phú hơn, dẫn đến khó khăn nhất định khi áp dụng cho dữ liệu tiếng Anh không dấu.

Dựa trên kết quả thực nghiệm, PARSeq là mô hình đạt hiệu suất cao nhất trên cả ba tập dữ liệu VietSignboard, English và Vin, đồng thời thể hiện sự vượt trội ổn định ở cả hai tiêu chí Exact-match và Normalized-match. Do đó, PARSeq được lựa chọn làm mô hình nhận dạng văn bản chính để tích hợp vào quy trình xử lý đầu-cuối (pipeline end-to-end) được đề xuất cho giai đoạn nhận dạng văn bản trên các vùng văn bản đã được xác định.

4.5.7 Kết quả đánh giá quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất phát hiện và nhận dạng văn bản trên biển hiệu

Sau khi tiến hành đánh giá độc lập và lựa chọn các mô hình tối ưu cho từng giai đoạn trong quy trình xử lý (pipeline), phần này trình bày kết quả đánh giá tổng thể hệ thống phát hiện và nhận dạng văn bản trên biển hiệu được đề xuất. Cụ thể, các mô hình được tích hợp vào quy trình xử lý (pipeline) dựa trên kết quả thực nghiệm của các giai đoạn

đã trình bày tại Mục 4.5.1, 4.5.5 và 4.5.6, trong đó mô hình có hiệu suất tốt nhất được lựa chọn cho mỗi thành phần tương ứng. Đối với giai đoạn phát hiện biển hiệu, việc lựa chọn mô hình được thực hiện trong từng nhóm phương pháp dựa trên dạng biểu diễn đầu ra, bao gồm vùng bao chữ nhật, vùng bao định hướng và phân đoạn đa giác, nhằm đảm bảo tính công bằng và nhất quán trong so sánh. Ngoài ra, quy trình xử lý (pipeline) cũng xem xét việc có và không áp dụng bước căn chỉnh biển hiệu đối với các mô hình có đầu ra định hướng hoặc đa giác, nhằm đánh giá ảnh hưởng của bước tiền xử lý này đến hiệu quả tổng thể của hệ thống. Trên cơ sở đó, quy trình xử lý (pipeline) hoàn chỉnh được xây dựng và đánh giá theo cách tiếp cận đầu cuối (end-to-end), nhằm phản ánh hiệu quả hoạt động của hệ thống trong bối cảnh thực tế của biển hiệu đường phố Việt Nam.

Đánh giá kết quả phát hiện văn bản trong quy trình xử lý đầu-cuối (pipeline end-to-end)

Bảng 4.11 trình bày kết quả đánh giá hiệu suất phát hiện văn bản của quy trình xử lý đầu-cuối (pipeline end-to-end) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText. Nhìn chung, các quy trình xử lý đầu-cuối (pipeline end-to-end) đều xuất đều đạt kết quả ổn định trên cả ba tập con, trong đó hiệu suất trên hai tập VietSignboard và Vin nổi bật hơn, phản ánh tính phù hợp và hiệu quả của hệ thống trong bối cảnh đường phố tại Việt Nam.

Đáng chú ý, quy trình xử lý đầu-cuối (pipeline end-to-end) kết hợp ba mô hình RTDETRv2, YOLOv8-OBB và PARSeq, tương ứng với các giai đoạn phát hiện biển hiệu, phát hiện văn bản và nhận dạng văn bản, đạt hiệu suất vượt trội so với các cấu hình quy trình xử lý đầu-cuối (pipeline end-to-end) còn lại. Cụ thể, quy trình xử lý đầu-cuối (pipeline end-to-end) này đạt chỉ số Precision lần lượt là **91.79**, **80.13** và **92.41** trên ba tập VietSignboard, English và Vin, cho thấy khả năng phát hiện chính xác phần lớn các vùng văn bản trong ảnh. Đồng thời, chỉ số Hmean của quy trình xử lý đầu-cuối (pipeline end-to-end) này đạt **89.64** trên VietSignboard và **89.79** trên Vin, khẳng định hiệu quả tổng thể của quy trình xử lý đầu-cuối (pipeline end-to-end) trong các trường hợp biển hiệu thực tế.

Trong khi đó, quy trình xử lý đầu-cuối (pipeline end-to-end) sử dụng mô hình SegFormer cho giai đoạn phát hiện biển hiệu kết hợp với YOLOv8-OBB và PARSeq

thể hiện ưu thế rõ rệt về chỉ số Recall. Pipeline này đạt Recall cao nhất hai tập English và Vin với các giá trị lần lượt là **80.25** và **87.44**. Đặc biệt, trên tập English, quy trình xử lý đầu-cuối (pipeline end-to-end) với SegFormer còn vượt trội quy trình xử lý đầu-cuối (pipeline end-to-end) sử dụng RTDETRv2 về chỉ số Hmean, đạt **77.72**, cho thấy khả năng phát hiện các vùng văn bản trong bối cảnh biển hiệu có bối cảnh phức tạp hoặc góc nhìn không thuận lợi.

Bên cạnh đó, khóa luận tiến hành đánh giá ảnh hưởng của bước căn chỉnh biển hiệu (signboard alignment) đối với các mô hình có đầu ra định hướng hoặc đa giác. Kết quả cho thấy, việc áp dụng căn chỉnh biển hiệu giúp cải thiện nhẹ các chỉ số Precision và Hmean trên hai tập VietSignboard và Vin, tuy nhiên mức cải thiện chưa thực sự đáng kể. Ngược lại, trên tập English, bước căn chỉnh có xu hướng làm giảm hiệu suất, đặc biệt ở các chỉ số Recall và Hmean, cho thấy cơ chế căn chỉnh không phải lúc nào cũng mang lại lợi ích trong các trường hợp biển hiệu có đặc điểm hình học đa dạng. Điều này phản ánh tính ổn định của mô hình YOLOv8-OBB trong cả hai thiết lập có và không áp dụng căn chỉnh biển hiệu. Hình [4.6](#) minh họa kết quả phát hiện văn bản trên biển hiệu trong hai trường hợp: không áp dụng và có áp dụng điều chỉnh biển hiệu với cơ chế perspective transformation (được trình bày tại Mục [4.2.1](#)). Có thể thấy rằng việc căn chỉnh giúp chuẩn hóa hình dạng vùng biển hiệu, từ đó cải thiện chất lượng dữ liệu đầu vào cho các bước xử lý phía sau.

Dựa trên phân tích kết quả có thể thấy rằng quy trình xử lý đầu-cuối (pipeline end-to-end) kết hợp RTDETRv2, YOLOv8-OBB và PARSeq là lựa chọn phù hợp khi triển khai trong bối cảnh đường phố Việt Nam, nhờ đạt độ chính xác cao và tính ổn định trên nhiều tập dữ liệu. Tuy nhiên, trong các tình huống thực tế mà biển hiệu thường bị nghiêng hoặc biến dạng mạnh theo góc nhìn camera, việc sử dụng SegFormer kết hợp với bước căn chỉnh biển hiệu cũng là một hướng tiếp cận tiềm năng. Ngoài ra, kết quả trong [Bảng 4.11](#) cho thấy khi giới hạn phạm vi phát hiện văn bản trong vùng biển hiệu, mô hình YOLOv8-OBB đạt độ chính xác cao hơn so với việc phát hiện trực tiếp trên toàn ảnh ngoại cảnh. Điều này cho thấy hiệu quả của bài toán khi tập trung phát hiện văn bản trong phạm vi biển hiệu, qua đó giảm ảnh hưởng của một số văn bản nhiễu hoặc



Không áp dụng căn chỉnh biển hiệu



Áp dụng căn chỉnh biển hiệu

Hình 4.6: So sánh trực quan kết quả phát hiện văn bản trên biển hiệu trong hai trường hợp không áp dụng và có áp dụng căn chỉnh biển hiệu (signboard alignment)

khó đọc, đồng thời vẫn bảo đảm thông tin quan trọng phục vụ cho các ứng dụng trích xuất thông tin và phân tích nội dung biển hiệu.

Đánh giá kết quả nhận dạng văn bản trong quy trình xử lý đầu-cuối (pipeline end-to-end)

Bảng 4.12 trình bày kết quả đánh giá hiệu suất nhận dạng văn bản theo cách tiếp cận đầu cuối (end-to-end) của các quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất trên ba tập con VietSignboard, English và Vin. Chỉ số Hmean_{e2e} được tính theo hai tiêu chí Exact-match và Normalized-match, qua đó phản ánh không chỉ hiệu quả của giai đoạn nhận dạng văn bản mà còn mức độ ảnh hưởng từ các bước phát hiện biển hiệu và phát hiện văn bản trước đó.

Dựa trên kết quả thực nghiệm, quy trình xử lý đầu-cuối (pipeline end-to-end) kết hợp RTDETRv2, YOLOv8-OBB và PARSeq đạt hiệu suất cao nhất trên hai tập VietSignboard và Vin, với Hmean_{e2e} lần lượt đạt **71.36%** (Exact-match) và **72.32%** (Normalized-match) trên tập VietSignboard, và **70.35%** (Exact-match) và **72.23%** (Normalized-match) trên tập Vin. Hiệu quả của các giai đoạn phát hiện đóng vai trò quan trọng trong việc giúp mô hình PARSeq khai thác tốt các vùng văn bản đầu vào, qua đó nâng cao độ chính xác của kết quả nhận dạng theo cách tiếp cận đầu-cuối (end-to-end).

Trong khi đó, trên tập English, quy trình xử lý đầu-cuối (pipeline end-to-end) sử dụng SegFormer cho giai đoạn phát hiện biển hiệu đạt hiệu suất tốt nhất, với Hmean_{e2e} đạt **48.92%** (Exact-match) và **49.67%** (Normalized-match), cho thấy SegFormer có khả năng thích ứng tốt hơn trong bối cảnh dữ liệu tiếng Anh. Bên cạnh đó, việc kết hợp bước

Bảng 4.11: Hiệu suất phát hiện văn bản của quy trình xử lý đầu-cuối (pipeline end-to-end) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model			Vietsignboard			English			Vin		
Signboard Det	Text Det	Text Rec	P	R	H	P	R	H	P	R	H
RTDETRv2	YOLOv8-OBB	PARSeq	91.79	87.59	89.64	80.13	<u>72.16</u>	<u>75.94</u>	92.41	<u>87.32</u>	89.79
YOLOv11-OBB	YOLOv8-OBB	PARSeq	89.34	86.76	88.03	73.47	67.54	70.38	89.39	85.59	87.45
SegFormer	YOLOv8-OBB	PARSeq	88.31	<u>87.30</u>	87.80	75.33	80.25	77.72	88.21	87.44	87.82
YOLOv11-OBB + Align	YOLOv8-OBB	PARSeq	89.78	86.79	<u>88.26</u>	73.48	67.41	70.32	89.25	85.33	87.25
SegFormer + Align	YOLOv8-OBB	PARSeq	90.64	85.52	88.00	<u>75.44</u>	71.18	73.25	<u>89.55</u>	86.32	<u>87.90</u>

Bảng 4.12: Hiệu suất nhận dạng văn bản của quy trình xử lý đầu-cuối (pipeline end-to-end) trên ba tập con VietSignboard, English và Vin của tập dữ liệu SignboardText. Chỉ số tốt nhất được đánh dấu đậm, chỉ số tốt thứ hai được gạch dưới.

Model			Hmean _{e2e} (%)					
Signboard Det	Text Det	Text Rec	Vietsignboard		English		Vin	
			Exact-match	Norm-match	Exact-match	Norm-match	Exact-match	Norm-match
RTDETRv2	YOLOv8-OBB	PARSeq	71.36	72.32	48.68	49.70	70.35	72.23
YOLOv11-OBB	YOLOv8-OBB	PARSeq	70.23	71.19	41.65	42.66	69.00	70.69
SegFormer	YOLOv8-OBB	PARSeq	69.94	70.94	48.92	49.67	69.30	70.91
YOLOv11-OBB + Align	YOLOv8-OBB	PARSeq	70.46	71.43	41.69	42.70	68.32	70.02
SegFormer + Align	YOLOv8-OBB	PARSeq	<u>70.25</u>	<u>71.19</u>	44.80	46.02	<u>70.12</u>	<u>71.65</u>

căn chỉnh biến hiệu mang lại cải thiện nhẹ trên hai tập VietSignboard và Vin, đặc biệt trong các trường hợp biến hiệu có góc nghiêng lớn hoặc hình dạng không chuẩn.

Kết luận và Lựa chọn quy trình xử lý đầu-cuối (pipeline end-to-end) Tối ưu Tổng hợp kết quả đánh giá phát hiện và nhận dạng văn bản trong quy trình xử lý đầu-cuối (pipeline end-to-end), có thể thấy rằng quy trình xử lý đầu-cuối (pipeline end-to-end) kết hợp RTDETRv2 cho phát hiện biến hiệu, YOLOv8-OBB cho phát hiện văn bản và PARSeq cho nhận dạng văn bản là lựa chọn cân bằng giữa độ chính xác và tính ổn định trên các tập dữ liệu thử nghiệm. Pipeline này đặc biệt phù hợp với bối cảnh biến hiệu đường phố Việt Nam, nơi chất lượng phát hiện đóng vai trò quan trọng trong việc bảo đảm hiệu quả của giai đoạn nhận dạng. Do đó, quy trình xử lý đầu-cuối (pipeline end-to-end) trên được lựa chọn cho hệ thống đề xuất trong khóa luận. Bên cạnh đó, trong các trường hợp biến hiệu có góc nghiêng lớn hoặc hình dạng không chuẩn, quy trình xử lý đầu-cuối (pipeline end-to-end) sử dụng SegFormer cho giai đoạn phát hiện biến hiệu kết hợp với bước căn chỉnh biến hiệu cũng cho thấy tiềm năng cải thiện hiệu suất nhận dạng, và có thể được xem là một hướng tiếp cận phù hợp trong những bối cảnh thách thức.

4.6 Ứng dụng minh họa quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất

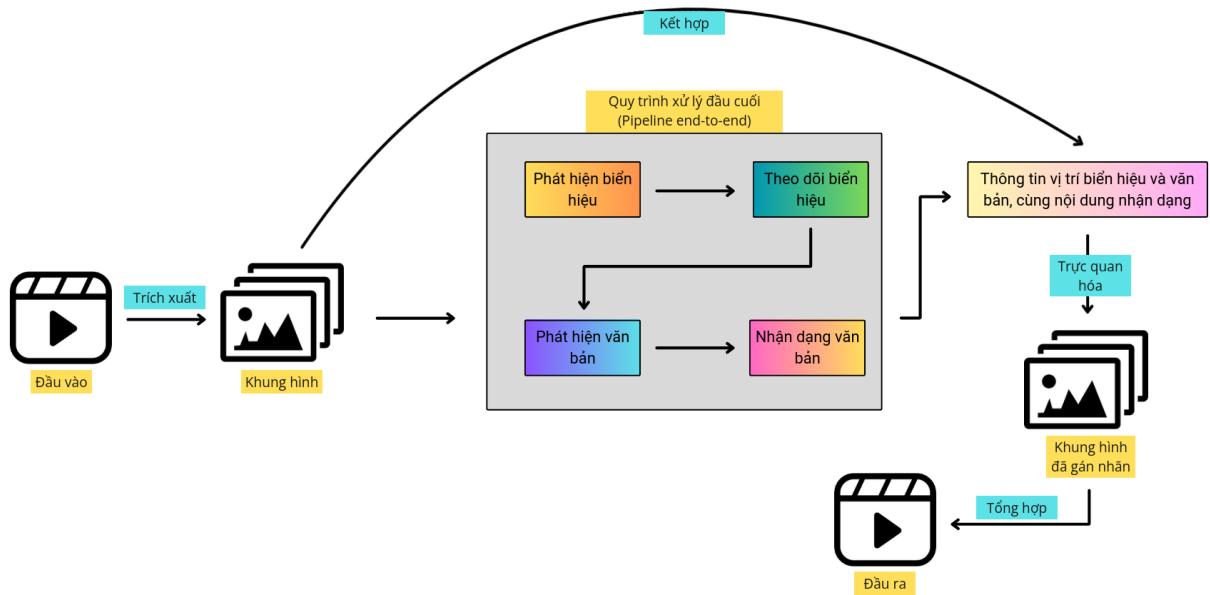
4.6.1 Mục tiêu và thiết kế ứng dụng minh họa

Nhằm đánh giá khả năng tổng quát hóa của quy trình phát hiện và nhận dạng văn bản trên biển hiệu đầu-cuối (end-to-end) đã được đề xuất trong bối cảnh dữ liệu thực tế, khóa luận xây dựng một ứng dụng minh họa thông qua việc xử lý các video thu thập từ môi trường đường phố Việt Nam. Ứng dụng này được sử dụng như một công cụ trực quan nhằm minh chứng tính khả thi và hiệu quả của quy trình khi áp dụng vào bối cảnh giao thông và cảnh quan đô thị tại Việt Nam.

Ứng dụng minh họa không hướng đến việc xây dựng một hệ thống hoàn chỉnh với giao diện người dùng hoặc các chức năng tương tác phức tạp. Thay vào đó, ứng dụng tập trung vào việc biểu diễn kết quả xử lý của pipeline dưới dạng video đầu ra đã được gán nhãn, nhằm phục vụ mục đích minh họa và đánh giá định tính trong phạm vi khóa luận. Cách tiếp cận này cho phép khóa luận tập trung vào việc xây dựng và đánh giá quy trình xử lý đầu-cuối (pipeline end-to-end) cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu.

Về mặt thiết kế tổng thể, ứng dụng được xây dựng theo quy trình xử lý tuần tự, trong đó dữ liệu video đầu vào được xử lý theo từng khung hình. Trên mỗi khung hình, pipeline lần lượt thực hiện các tác vụ phát hiện biển hiệu, phát hiện vùng văn bản trong biển hiệu và nhận dạng nội dung văn bản. Kết quả xử lý được tổng hợp và trực quan hóa trực tiếp trên các khung hình tương ứng, sau đó được tái cấu trúc thành video đầu ra. Hình 4.7 minh họa kiến trúc tổng thể của quy trình xử lý trong ứng dụng minh họa.

Đầu ra của ứng dụng là các video đã được xử lý và gán nhãn, cho phép quan sát trực tiếp kết quả suy luận của hệ thống. Thông qua video đầu ra, người xem có thể đánh giá một cách định tính khả năng phát hiện biển hiệu, mức độ chính xác của các vùng văn bản được xác định, cũng như chất lượng nhận dạng nội dung văn bản trong các điều kiện thực tế khác nhau.



Hình 4.7: Sơ đồ quy trình xử lý video đầu-cuối (end-to-end) trong ứng dụng minh họa cho bài toán phát hiện và nhận dạng văn bản trên biến hiệu.

4.6.2 Triển khai quy trình xử lý đầu-cuối (pipeline end-to-end) để xuất trên video

Dựa trên kiến trúc tổng thể đã được trình bày ở 4.6.1, phần này trình bày chi tiết cách thức triển khai quy trình xử lý đầu-cuối (end-to-end pipeline) trên dữ liệu video thu thập từ môi trường đường phố Việt Nam. Việc triển khai được thực hiện theo hình thức xử lý video ngoại tuyến, trong đó pipeline suy luận được áp dụng tuần tự trên từng khung hình của video đầu vào.

Các video đầu vào được tiếp nhận dưới nhiều định dạng phổ biến và được xử lý theo cơ chế từng khung hình. Ở mỗi bước xử lý, hệ thống lần lượt trích xuất khung hình từ video, đồng thời thu thập các thông tin cần thiết như tốc độ khung hình và độ phân giải để đảm bảo tính nhất quán khi tái tạo video đầu ra. Cách tiếp cận xử lý theo từng khung hình này cho phép pipeline học sâu, vốn được thiết kế chủ yếu cho dữ liệu ảnh tĩnh, có thể được áp dụng trực tiếp lên dữ liệu video mà không cần thay đổi đáng kể về mặt kiến trúc.

Trước khi đưa vào pipeline suy luận, mỗi khung hình được thực hiện một số bước tiền xử lý nhằm đảm bảo tương thích với yêu cầu đầu vào của các mô hình học sâu. Cụ

thể, khung hình được chuyển đổi không gian màu phù hợp, chuẩn hóa kích thước và giá trị điểm ảnh, đồng thời được biểu diễn dưới dạng tensor để phục vụ cho quá trình suy luận. Các bước tiền xử lý này giúp giảm sự sai lệch dữ liệu và đảm bảo tính ổn định của kết quả khi áp dụng pipeline trên các video có điều kiện ánh sáng và độ phân giải khác nhau.

Trên mỗi khung hình đã được tiền xử lý, quy trình xử lý đầu-cuối (pipeline end-to-end) thực hiện theo trình tự gồm ba giai đoạn chính.

- **Phát hiện biến hiệu:** Quy trình xử lý (pipeline) xác định các vùng biến hiệu xuất hiện trong khung hình và trích xuất chúng từ hình gốc, đóng vai trò là vùng quan tâm (Region of Interest - ROI) cho các bước tiếp theo.
- **Phát hiện văn bản:** Trong mỗi ROI, Quy trình xử lý (pipeline) xác định vị trí các vùng văn bản nhằm xác định chính xác không gian của văn bản.
- **Nhận dạng văn bản:** Các vùng văn bản được trích xuất, chuẩn hóa hình dạng và đưa vào mô-đun nhận dạng để suy luận nội dung.

Bên cạnh đó, nhằm đảm bảo tính nhất quán của thông tin trên chuỗi khung hình liên tiếp, hệ thống tích hợp cơ chế theo dõi biến hiệu. Việc theo dõi cho phép gán định danh cho các biến hiệu xuất hiện xuyên suốt video, qua đó hạn chế việc nhận dạng lặp lại cùng một đối tượng và hỗ trợ tổng hợp thông tin từ nhiều khung hình khác nhau, song, trong phạm vi của khóa luận, cơ chế theo dõi chỉ đóng vai trò hỗ trợ và không phải là trọng tâm chính của pipeline đề xuất.

Sau quá trình suy luận, các thông tin về vị trí biến hiệu, vùng văn bản và nội dung nhận dạng được kết hợp với khung hình gốc và trực quan hóa lên từng khung hình. Đối với văn bản tiếng Việt, hệ thống sử dụng phông chữ hỗ trợ Unicode nhằm đảm bảo biểu diễn đầy đủ các ký tự có dấu. Các khung hình đã gán nhãn được tổng hợp lại theo đúng thứ tự thời gian ban đầu để tạo thành video đầu ra hoàn chỉnh, cho phép quan sát trực tiếp kết quả suy luận của hệ thống.

Các bước xử lý và trực quan hóa video đầu ra được thực hiện trên nền tảng Python, kết hợp giữa các thư viện xử lý ảnh/video truyền thống như OpenCV, Matplotlib, và các



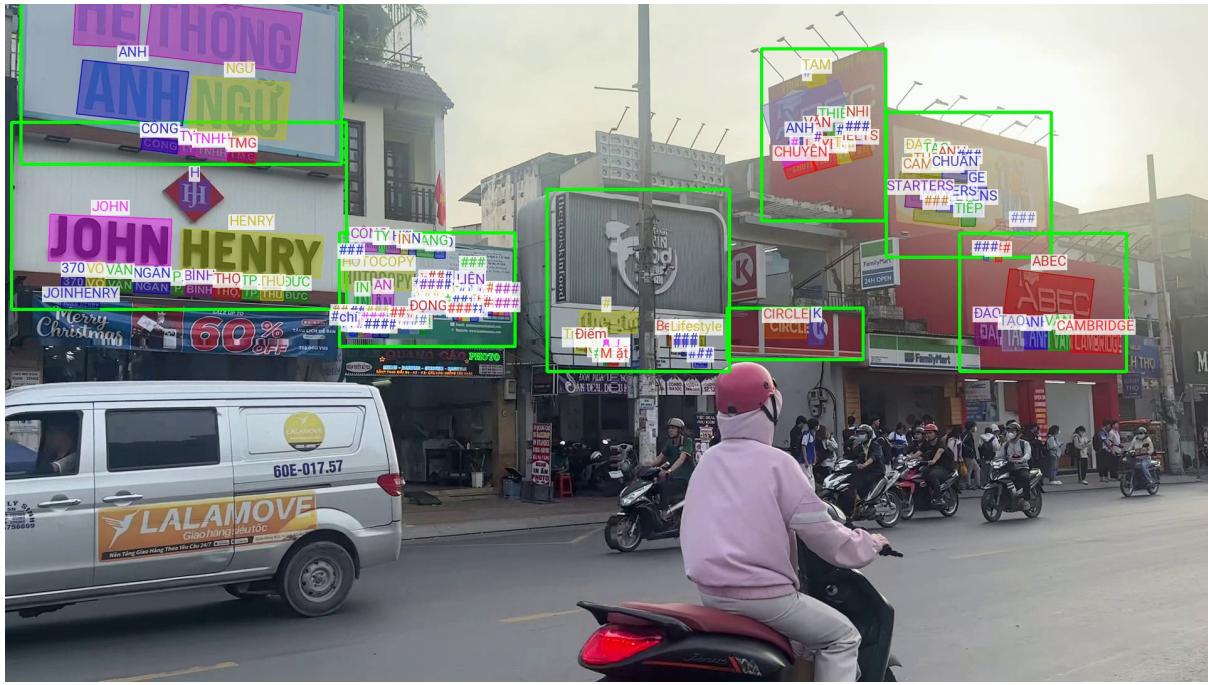
Hình 4.8: Kết quả suy luận của quy trình xử lý đầu-cuối (pipeline end-to-end) trong điều kiện quan sát bất lợi, với các biển hiệu ở xa và sử dụng kiểu chữ nghệ thuật.

khung làm việc (framework) học sâu hiện đại như PyTorch, cùng các thư viện hỗ trợ tương ứng.

4.6.3 Kết quả minh họa và đánh giá định tính

Để đánh giá định tính khả năng suy luận của quy trình xử lý đầu-cuối (pipeline end-to-end) trên dữ liệu video, khóa luận tiến hành áp dụng quy trình xử lý (pipeline) để xuất lén các video thực tế được thu thập trong bối cảnh đường phố Việt Nam. Các video được ghi nhận tại một số khu vực đô thị tiêu biểu ở Thành phố Hồ Chí Minh (ví dụ: khu vực đường Lê Văn Việt, Võ Văn Ngân), phản ánh đa dạng điều kiện quan sát trong môi trường thực tế. Từ kết quả suy luận, khóa luận lựa chọn một số khung hình tiêu biểu các thời điểm khác nhau trong video nhằm minh họa và phân tích định tính khả năng phát hiện biển hiệu, xác định vùng văn bản và nhận dạng nội dung văn bản của quy trình xử lý (pipeline).

Hình 4.8 minh họa kết quả suy luận của quy trình xử lý đầu-cuối (pipeline end-to-end). Kết quả cho thấy pipeline có khả năng phát hiện tốt đối với các biển hiệu nằm ở



Hình 4.9: Kết quả suy luận của pipeline trong điều kiện quan sát thuận lợi, khi camera tiến gần hơn tới khu vực các biển hiệu.

vị trí gần và có kích thước đủ lớn. Tuy nhiên, đối với các biển hiệu ở xa hoặc có kích thước nhỏ, khả năng phát hiện còn hạn chế. Hơn nữa, việc nhận dạng văn bản cũng gặp khó khăn khi biển hiệu sử dụng kiểu chữ nghệ thuật hoặc được cách điệu. Điều này cho thấy hiệu suất của pipeline vẫn chịu ảnh hưởng đáng kể bởi các yếu tố như khoảng cách quan sát, kích thước đối tượng, và kiểu trình bày văn bản trong môi trường thực tế.

Trong điều kiện quan sát thuận lợi hơn, Hình 4.9 thể hiện kết quả suy luận tại thời điểm camera tiến gần hơn tới khu vực các biển hiệu. Trong trường hợp này, pipeline hoạt động tương đối hiệu quả khi phát hiện được đa số các biển hiệu xuất hiện trong khung hình, đồng thời xác định chính xác các vùng văn bản bên trong. Kết quả nhận dạng cho thấy hệ thống có khả năng suy luận tốt đối với cả văn bản tiếng Việt có dấu và một số văn bản tiếng Anh trên biển hiệu. Kết quả này phản ánh khả năng tổng quát hóa tương đối tốt của pipeline trong bối cảnh đường phố Việt Nam khi điều kiện quan sát thuận lợi hơn.

Bên cạnh các trường hợp trên, Hình 4.10 minh họa một trường hợp biển hiệu có sự xuất hiện của các thành phần đồ họa như biểu tượng hoặc logo xen kẽ với văn bản. Mặc



Hình 4.10: Kết quả suy luận của pipeline trong trường hợp biển hiệu có sự xuất hiện của các thành phần phi văn bản (biểu tượng, logo) xen kẽ với nội dung văn bản.

dù các thành phần này không mang thông tin ngôn ngữ, pipeline vẫn tập trung chủ yếu vào các vùng chứa văn bản và phát hiện đúng phần lớn các dòng chữ chính. Điều này cho thấy hệ thống có khả năng nhận diện các vùng chứa thông tin văn bản ngay cả khi tồn tại các thành phần phi văn bản trong cùng một biển hiệu.

Nhìn chung, các kết quả minh họa cho thấy quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất hoạt động hiệu quả trong việc phát hiện và nhận dạng văn bản trên biển hiệu dưới nhiều điều kiện quan sát khác nhau trong môi trường đường phố Việt Nam. Tuy nhiên, bên cạnh những trường hợp quy trình xử lý (pipeline) cho kết quả khả quan, một số thách thức vẫn tồn tại khi biển hiệu ở xa, sử dụng kiểu chữ nghệ thuật hoặc chứa văn bản có kích thước lớn, ảnh hưởng đến hiệu quả phát hiện và nhận dạng văn bản. Những quan sát định tính này cung cấp cơ sở cho các hướng cải thiện trong tương lai của khóa luận.

Chương 5

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Trước nhu cầu ngày càng tăng trong việc khai thác thông tin từ cảnh quan đô thị, biển hiệu trở thành nguồn dữ liệu văn bản giàu ngữ nghĩa, cung cấp thông tin quan trọng cho các hệ thống thị giác máy tính. Chính vì vậy, việc tự động phát hiện và trích xuất nội dung từ các biển hiệu đóng vai trò quan trọng trong việc chuyển đổi dữ liệu thô thành tri thức có thể khai thác, từ đó hỗ trợ các nhiệm vụ trích xuất thông tin một cách hiệu quả. Trên cơ sở đó, khóa luận tập trung xây dựng một quy trình xử lý đầu-cuối (pipeline end-to-end), cung cấp đồng thời thông tin về vị trí và nội dung văn bản trên biển hiệu.

Xuất phát từ mục tiêu trên, ba đóng góp chính của khóa luận bao gồm:

- **Mở rộng bộ dữ liệu:** Bổ sung nhãn đối tượng biển hiệu (signboard bounding box) cho tập dữ liệu ảnh tĩnh SignboardText, đồng thời thu thập một tập dữ liệu video hành trình thực tế, phục vụ minh họa và kiểm tra tính tổng quát của quy trình xử lý (pipeline).
- **Thực nghiệm và đánh giá:** Tiến hành cài đặt, thực nghiệm và phân tích kết quả cho các tác vụ phát hiện biển hiệu, phát hiện văn bản và nhận dạng văn bản, qua đó làm rõ ưu và nhược điểm của các phương pháp trong bối cảnh dữ liệu đường phố Việt Nam, đồng thời cung cấp cơ sở tham khảo cho các nghiên cứu và ứng dụng trong tương lai.

- **Xây dựng quy trình xử lý đầu-cuối (pipeline end-to-end):** Xây dựng quy trình xử lý đầu-cuối hoàn chỉnh cho bài toán phát hiện và nhận dạng văn bản trên biển hiệu trong video hành trình tại Việt Nam.

Hiệu quả của quy trình xử lý đầu-cuối (pipeline end-to-end) đề xuất đã được đánh giá trên tập SignboardText, với khả năng xử lý tốt các biển hiệu chứa phần lớn ngôn ngữ Tiếng Việt. Nhờ đó, quy trình xử lý (pipeline) không chỉ hỗ trợ các ứng dụng trích xuất thông tin trong thực tế mà còn cung cấp công cụ hữu ích cho việc thu thập và quản lý dữ liệu trong lĩnh vực thị giác máy tính. Tuy nhiên, trong quá trình thực nghiệm và đánh giá, khóa luận nhận thấy một số hạn chế đáng lưu ý:

- **Điều kiện môi trường:** Thực nghiệm chủ yếu được thực hiện trong các điều kiện thuận lợi, do đó hiệu quả của quy trình xử lý đầu-cuối (pipeline end-to-end) trong các môi trường bất lợi như mưa, sương mù hoặc ánh sáng yếu vẫn chưa được kiểm chứng đầy đủ.
- **Dữ liệu đánh giá:** Mặc dù bước tiền xử lý căn chỉnh biển hiệu giúp cải thiện hiệu quả của quy trình xử lý (pipeline), tập dữ liệu SignboardText vẫn còn hạn chế về số lượng biển hiệu có các góc nghiêng đa dạng. Bên cạnh đó, do chưa có tập dữ liệu video đường phố Việt Nam được gán nhãn đầy đủ (xuất phát từ hạn chế về tài nguyên gán nhãn), khóa luận hiện chưa thể thực hiện đánh giá định lượng toàn diện quy trình xử lý đầu-cuối (pipeline end-to-end) trên dữ liệu video. Điều này làm hạn chế khả năng kiểm chứng đầy đủ tính tổng quát hóa của quy trình xử lý (pipeline) trong bối cảnh video đường phố thực tế.

5.2 Hướng phát triển

Nhằm khắc phục những hạn chế đã nêu và nâng cao hiệu quả của quy trình xử lý đầu-cuối (pipeline end-to-end) trong các điều kiện thực tế đa dạng, khóa luận đề xuất một số hướng phát triển trong tương lai như sau:

- **Cải thiện dữ liệu:** Xây dựng bộ dữ liệu chuyên biệt chứa các biển hiệu nghiêng ở nhiều góc độ khác nhau, đồng thời mở rộng việc thu thập và gán nhãn dữ liệu

video đường phố Việt Nam. Việc bổ sung dữ liệu video được gán nhãn đầy đủ sẽ tạo điều kiện cho việc đánh giá định lượng toàn diện hơn hiệu quả của quy trình xử lý đầu-cuối (pipeline end-to-end), qua đó góp phần cải thiện và kiểm chứng khả năng tổng quát hóa của quy trình xử lý (pipeline) trong các bối cảnh thực tế.

- **Mở rộng điều kiện:** Thực nghiệm quy trình xử lý đầu-cuối (pipeline end-to-end) trong các môi trường bất lợi, bao gồm mưa, ánh sáng yếu và sương mù, để kiểm chứng tính ổn định và khả năng triển khai thực tế.
- **Ứng dụng thực tế:** Phát triển giao diện người dùng cho ứng dụng phát hiện và nhận dạng văn bản trên biển hiệu, hỗ trợ phân tích biển hiệu, tra cứu địa điểm và thông tin cửa hàng tại Việt Nam.

Tài liệu tham khảo

- [1] Rowel Atienza. Vision transformer for fast and efficient scene text recognition. In *International conference on document analysis and recognition*, pages 319–334. Springer, 2021. [6](#), [8](#), [9](#), [17](#), [31](#), [32](#)
- [2] Youngmin Baek, Bado Lee, Dongyo Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9365–9374, 2019. [15](#)
- [3] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. In *European conference on computer vision*, pages 178–196. Springer, 2022. [6](#), [8](#), [9](#), [17](#), [32](#), [33](#)
- [4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. [6](#), [7](#), [8](#), [13](#), [23](#), [24](#)
- [5] Zhe Chen, Jiahao Wang, Wenhai Wang, Guo Chen, Enze Xie, Ping Luo, and Tong Lu. Fast: Faster arbitrarily-shaped text detector with minimalist kernel representation. *arXiv preprint arXiv:2111.02394*, 2021. [6](#), [7](#), [8](#), [15](#), [29](#), [30](#)
- [6] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. [6](#), [7](#), [8](#), [25](#), [27](#)

- [7] T. Do, T. Tran, T. Nguyen, D.-D. Le, and T. D. Ngo. Signboardtext: Text detection and recognition in in-the-wild signboard images. *IEEE Access*, 12:62942–62957, 2024. [3](#), [4](#), [6](#), [8](#), [16](#), [19](#), [41](#), [42](#), [43](#), [44](#)
- [8] Yongkun Du, Zhineng Chen, Caiyan Jia, Xieping Gao, and Yu-Gang Jiang. Out of length text recognition with sub-string matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 2798–2806, 2025. [6](#), [8](#), [9](#), [17](#), [34](#), [35](#)
- [9] Yongkun Du, Zhineng Chen, Caiyan Jia, Xiaoting Yin, Tianlun Zheng, Chenxia Li, Yuning Du, and Yu-Gang Jiang. Svtr: Scene text recognition with a single visual model. *arXiv preprint arXiv:2205.00159*, 2022. [34](#)
- [10] Yongkun Du, Zhineng Chen, Hongtao Xie, Caiyan Jia, and Yu-Gang Jiang. Svtrv2: Ctc beats encoder-decoder models in scene text recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20147–20156, 2025. [6](#), [8](#), [9](#), [17](#), [34](#), [36](#)
- [11] Evezerest. PPOCRLLabel: Semi-automatic image annotation tool for ocr. GitHub repository, 2023. Available: <https://github.com/Evezerest/PPOCRLLabel>, accessed Sep. 10, 2025. [43](#)
- [12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. [12](#)
- [13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. [12](#)
- [14] Xu Han, Junyu Gao, Chuang Yang, Yuan Yuan, and Qi Wang. Spotlight text detector: Spotlight on candidate regions like a camera. *IEEE Transactions on Multimedia*, 2024. [14](#)

- [15] JianJun Kang, Mayire Ibrayim, and Askar Hamdulla. Overview of scene text detection and recognition. In *Proceedings of the 14th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 661–666, 2022. [14](#), [16](#), [19](#), [26](#), [31](#)
- [16] Taeho Kil, Seonghyeon Kim, Sukmin Seo, Yoonsik Kim, and Daehee Kim. Towards unified scene text spotting based on sequence generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15223–15232, 2023. [6](#), [8](#), [9](#), [20](#), [38](#), [39](#)
- [17] Chae Young Lee, Youngmin Baek, and Hwalsuk Lee. Tedeval: A fair evaluation metric for scene text detectors. In *2019 international conference on document analysis and recognition workshops (ICDARW)*, volume 7, pages 14–17. IEEE, 2019. [51](#), [52](#)
- [18] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. Textboxes: A fast text detector with a single deep neural network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017. [14](#), [19](#)
- [19] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11474–11481, 2020. [27](#)
- [20] Minghui Liao, Zhisheng Zou, Zhaoyi Wan, Cong Yao, and Xiang Bai. Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):919–931, 2022. [6](#), [7](#), [8](#), [15](#), [27](#), [28](#), [29](#)
- [21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. [12](#)

- [22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. [12](#)
- [23] Xingtong Liu, Dawen Liang, Shi Yan, Dong Chen, Yu Qiao, and Junjie Yan. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, 2018. [18](#)
- [24] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, and Liangwei Wang. Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9809–9818, 2020. [14, 19](#)
- [25] Shangbang Long, Jiaqiang Ruan, Wenjie Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 20–36, 2018. [15](#)
- [26] Wenyu Lv, Yian Zhao, Qinyao Chang, Kui Huang, Guanzhong Wang, and Yi Liu. Rt-detr2: Improved baseline with bag-of-freebies for real-time detection transformer. *arXiv preprint arXiv:2407.17140*, 2024. [6, 7, 8, 24](#)
- [27] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 67–83, 2018. [16, 19](#)
- [28] Fatemeh Naiemi, Vahid Ghods, and Hassan Khalesi. Scene text detection and recognition: A survey. *Multimedia Tools and Applications*, 81(1):20255–20290, 2022. [14, 16, 19, 26, 31](#)
- [29] Rafael Padilla. Object-detection-metrics. GitHub repository, 2021. Available: <https://github.com/rafaelpadilla/Object-Detection-Metrics>, accessed Sep. 29, 2025. [51](#)

- [30] Umapada Pal, Arnab Halder, Palaiahnakote Shivakumara, and Michael Blumenstein. A comprehensive review on text detection and recognition in scene images. *Artificial Intelligence and Applications*, 2(4):229–249, 2024. [14](#), [16](#), [19](#), [26](#), [31](#)
- [31] Qian Qiao, Yu Xie, Jun Gao, Tianxiang Wu, Shaoyao Huang, Jiaqing Fan, Ziqiang Cao, Zili Wang, and Yue Zhang. Dntextspotter: Arbitrary-shaped scene text spotting via improved denoising training. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 10134–10143, 2024. [6](#), [8](#), [9](#), [20](#), [39](#), [40](#)
- [32] D. L. Quang, K. V. Sy, H. L. Viet, S. P. Bao, and H. B. Quang. Signboards detection from street-view image using convolutional neural network: A case study in vietnam. In *Proceedings of the RIVF International Conference on Computing and Communication Technologies (RIVF)*, pages 394–397, Ho Chi Minh City, Vietnam, 2022. [2](#)
- [33] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. [12](#), [22](#), [23](#)
- [34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. [12](#)
- [35] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016. [17](#)
- [36] Hans Thisanke, Chamli Deshan, Kavindu Chamith, Sachith Seneviratne, Rajith Vidanaarachchi, and Damayanthi Herath. Semantic segmentation using vision

transformers: A survey. *Engineering Applications of Artificial Intelligence*, 126:106669, 2023. [25](#)

- [37] Ultralytics. Yolo. <https://github.com/ultralytics/ultralytics>, 2023. [6](#), [7](#), [8](#), [51](#)
- [38] Analytics Vidhya. Yolo: An ultimate solution to object detection and classification, 2022. Accessed: 2026-01-10. [12](#)
- [39] Pengfei Wang, Chengquan Zhang, Fei Qi, Shanshan Liu, Xiaoqiang Zhang, Pengyuan Lyu, Junyu Han, Jingtuo Liu, Errui Ding, and Guangming Shi. Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2782–2790, 2021. [20](#)
- [40] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8440–8449, 2019. [6](#), [7](#), [8](#), [15](#), [26](#), [28](#)
- [41] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021. [6](#), [7](#), [8](#), [25](#), [26](#)
- [42] Jian Ye, Zhe Chen, Juhua Liu, and Bo Du. Textfusenet: Scene text detection with richer fused features. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 516–522, 2020. [16](#)
- [43] Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Tongliang Liu, Bo Du, and Dacheng Tao. Deepsolo: Let transformer decoder with explicit points solo for text spotting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19348–19357, 2023. [6](#), [8](#), [9](#), [20](#), [37](#), [38](#)

- [44] Shi-Xue Zhang, Xiaobin Zhu, Lei Chen, Jie-Bo Hou, and Xu-Cheng Yin. Arbitrary shape text detection via segmentation with probability maps. *IEEE transactions on pattern analysis and machine intelligence*, 45(3):2736–2750, 2022. [6](#), [7](#), [8](#), [15](#), [28](#)
- [45] Shi-Xue Zhang, Xiaobin Zhu, Jie-Bo Hou, Chang Liu, Chun Yang, Hongfa Wang, and Xu-Cheng Yin. Deep relational reasoning graph network for arbitrary shape text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9699–9708, 2020. [15](#)
- [46] Shi-Xue Zhang, Xiaobin Zhu, Jie-Bo Hou, Chun Yang, and Xu-Cheng Yin. Kernel proposal network for arbitrary shape text detection. *IEEE transactions on neural networks and learning systems*, 34(11):8731–8742, 2022. [6](#), [7](#), [8](#), [15](#), [30](#), [31](#)
- [47] Xiang Zhang, Yongwen Su, Subarna Tripathi, and Zhuowen Tu. Text spotting transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9519–9528, 2022. [6](#), [8](#), [9](#), [20](#), [36](#), [37](#)
- [48] Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16965–16974, 2024. [24](#)
- [49] Tianlun Zheng, Zhenpeng Chen, Shancheng Fang, Hongtao Xie, and Yu-Gang Jiang. Cdistsnet: Perceiving multi-domain character distance for robust text recognition. *International Journal of Computer Vision*, 132(2):300–318, 2024. [6](#), [8](#), [9](#), [17](#), [33](#), [34](#)
- [50] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017. [14](#)

- [51] Yiqin Zhu, Jianyong Chen, Lingyu Liang, Zhanghui Kuang, Lianwen Jin, and Wayne Zhang. Fourier contour embedding for arbitrary-shaped text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3123–3131, 2021. [14](#)
- [52] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023. [11](#), [22](#)