TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN   
VÀ TRUYỀN THÔNG VIỆT - HÀN

**KHOA KHOA HỌC MÁY TÍNH**



BÁO CÁO KỸ THUẬT

**Scene text recognition and License-plate**

Học phần: Thị giác máy tính

Sinh viên thực hiện: Trần Quang Minh – 22AD030

**Da Nang, May – 2025**

**MỤC LỤC**

Contents

[1.Giới thiệu 1](#_Toc198028319)

[2.Kiến trúc tổng thể của Hệ thống 1](#_Toc198028320)

[2.1. Giai đoạn Huấn luyện 1](#_Toc198028321)

[2.2 Giai đoạn Triển khai (Inference) 2](#_Toc198028322)

[3. Dữ liệu và tiền xử lý 2](#_Toc198028323)

[3.1.Các tập dữ liệu 2](#_Toc198028324)

[ICDAR2003 2](#_Toc198028325)

[License Plate Dataset 2](#_Toc198028326)

[3.2 Tiền xử lý dữ liệu 3](#_Toc198028327)

[4. Kiến trúc mô hình 4](#_Toc198028328)

[4.1 YOLO11 4](#_Toc198028329)

[4.2 CRNN 5](#_Toc198028330)

[5.Huấn luyện mô hình 6](#_Toc198028331)

[6. Kết quả thử nghiệm, so sánh, đánh giá 6](#_Toc198028332)

[Kết quả đánh giá mô hình CRNN trên tập test ICDAR2003 7](#_Toc198028333)

[7. Kết luận 8](#_Toc198028334)

[Ưu điểm: 8](#_Toc198028335)

[Nhược điểm: 8](#_Toc198028336)

[Tổng kết: 8](#_Toc198028337)

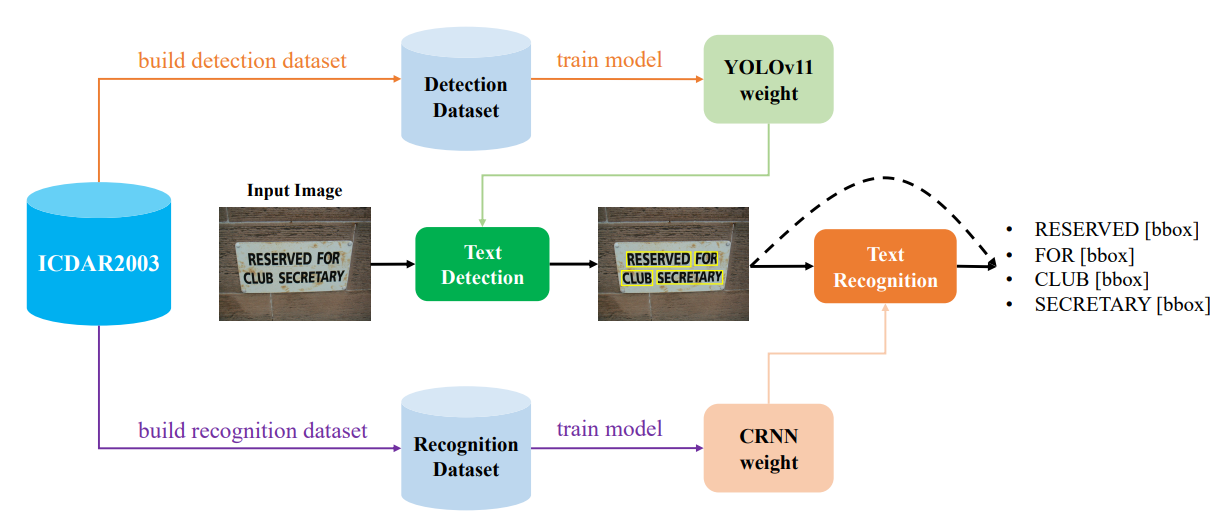
# 1.Giới thiệu

Mục đích của đề tài là xây dựng một hệ thống **Nhận dạng Văn bản trong Ảnh (SceneText Recognition - STR)** theo phương pháp **hai giai đoạn**, áp dụng vào bài toán thực tế. Dự án tập trung phát triển và đánh giá pipeline gồm mô hình **YOLOv11** cho phát hiện văn bản và **CRNN kết hợp CTC Loss** cho nhận dạng.

Các nội dung chính đã thực hiện gồm:

* Thiết kế pipeline gồm hai mô-đun: Text Detection và Text Recognition.
* Phân tích dữ liệu (EDA) trên hai tập: **ICDAR2003** và **biển số xe** để đánh giá bias và phân bố bounding box.
* Áp dụng kỹ thuật **data augmentation** nhằm cải thiện khả năng tổng quát hóa mô hình.
* Huấn luyện mô hình **YOLOv11** và **CRNN**, đồng thời so sánh hiệu năng với các mô hình khác như **EfficientDet** và **Faster R-CNN**.
* Đánh giá hệ thống qua các chỉ số loss, tốc độ xử lý, và mức độ phù hợp triển khai thực tế.
* Tổng kết ưu và nhược điểm của hệ thống STR đề xuất.

# 2.Kiến trúc tổng thể của Hệ thống



## ****2.**1. Giai đoạn Huấn luyện**

* **Chuẩn bị dữ liệu:**
  + Detection Dataset: Chuyển annotation từ XML (ICDAR2003) sang định dạng YOLO (center\_x, center\_y, w\_norm, h\_norm).
  + Recognition Dataset: Crop từng vùng văn bản từ ảnh, gán nhãn tương ứng (chuỗi ký tự).
* **Huấn luyện mô hình:**
  + **YOLOv11** → Huấn luyện trên Detection Dataset → tạo yolov11.pt để phát hiện văn bản.
  + **CRNN + CTC** → Huấn luyện trên Recognition Dataset → tạo crnn.pth để nhận dạng nội dung.

## **2.**2** Giai đoạn Triển khai (Inference)**

* Nhận ảnh đầu vào bất kỳ (chứa văn bản).
* **Text Detection** (YOLOv11): Trả về bounding boxes + độ tin cậy.
* **Text Recognition** (CRNN): Crop vùng chữ và chuyển thành chuỗi ký tự.
* **Output**: Danh sách từ + tọa độ tương ứng → hiển thị lên ảnh hoặc xuất JSON.

# 3. Dữ liệu và tiền xử lý

## 3.1.Các tập dữ liệu

ICDAR2003:Bộ dữ liệu được em tự thu thập từ nhiều nguồn ở trên kaggle EDA dữ liệu:

Dựa trên các biểu đồ EDA em kết luận như sau :

* **Chiều dài và chiều rộng**
  + Phần lớn các bounding box có chiều dài dưới 400 px và chiều rộng dưới 100 px.
  + Một số ít trường hợp ngoại lệ có bounding box rất dài hoặc rộng do văn bản nhiều ký tự.
* **Diện tích và tỷ lệ so với ảnh**
  + Diện tích hầu hết dưới 100,000 px².
  + Tỷ lệ diện tích bbox/ảnh thường < 10%, cho thấy văn bản chiếm phần nhỏ trong ảnh tổng thể.
* **Tọa độ trung tâm (X, Y)**
  + Phân bố khá đều dọc theo cả trục X và Y, nhưng tập trung nhẹ ở khoảng giữa ảnh (X≈400, Y≈200–300).
* **Phân bố 2 chiều tọa độ trung tâm**
  + Một cụm tập trung rõ ràng ở khu vực (X≈350–400, Y≈200–300), nhưng không bị lệch hẳn về một phía quá nhiều.
  + Điều này cho thấy có xu hướng **tụ trung tâm nhẹ**, nhưng không đủ để gây thiên lệch đáng kể.

License Plate Dataset: Bộ dữ liệu được em tự thu thập từ nhiều nguồn ở trên kaggle<https://www.kaggle.com/datasets/duchocgioi/license-plate-dataset>

EDA dữ liệu:

Dựa trên các biểu đồ EDA em kết luận như sau :

 **Chiều dài & chiều rộng bbox**:

* Chiều dài phần lớn nằm trong khoảng **100–200 px**.
* Chiều rộng đa số dưới **100 px**, thể hiện kích thước biển số nhỏ và đồng nhất.

 **Diện tích & tỷ lệ diện tích bbox/ảnh**:

* Diện tích phần lớn dưới **50,000 px²**.
* Tỷ lệ diện tích so với ảnh thường < **10%**, cho thấy biển số chiếm phần nhỏ trong ảnh.

 **Tọa độ trung tâm bbox (X, Y)** :

* **X-center** tập trung mạnh quanh khoảng **200–300 px**.
* **Y-center** tập trung khoảng **150–250 px**, cho thấy vị trí biển số thường nằm ở vùng dưới và gần giữa ảnh.

 **Phân bố 2D tọa độ trung tâm** :

* Rất rõ cụm bias nằm ở vùng **góc trên–trái** của ảnh (X≈300–400 px, Y≈200–300 px).
* Kích thước ảnh trung bình: **1600×1200 px**, cho thấy vị trí biển số chiếm khoảng:
  + **20–25% chiều rộng** → lệch trái trung tâm.
  + **15–25% chiều cao** → ở phía trên trung tâm.

## 3.2 Tiền xử lý dữ liệu

Trong quá trình huấn luyện, YOLO đã tích hợp sẵn một loạt các kỹ thuật **data augmentation** được **tự động kích hoạt** nhằm tăng tính đa dạng của dữ liệu và cải thiện khả năng tổng quát hóa của mô hình. Dưới đây là danh sách các augmentation mặc định được áp dụng trong YOLO cùng với mô tả và xác suất tương ứng:

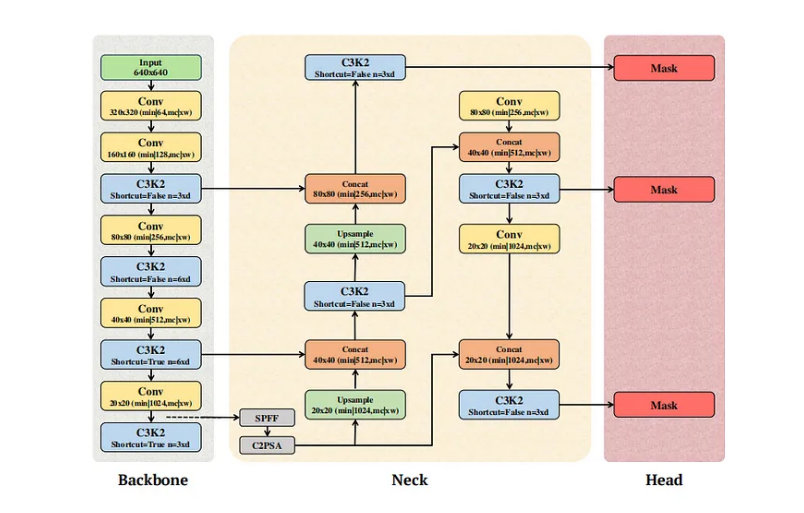
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên Augmentation** | **Mô tả chi tiết** | **Xác suất (p)** |
| 1 | Letterbox | Resize cạnh dài nhất về 640px, pad bằng màu (114,114,114) để ra ảnh 640x640 | Always |
| 2 | HorizontalFlip | Lật ngang ảnh (trái ↔ phải) | 0.5 |
| 3 | VerticalFlip | Lật dọc ảnh (trên ↔ dưới) | 0.2 |
| 4 | HueSaturationValue | Hue ±15, Saturation ±25, Value ±15 | 0.5 |
| 5 | RandomBrightnessContrast | Độ sáng ±20%, tương phản ±20% | 0.5 |
| 6 | ShiftScaleRotate | Dịch ±10%, zoom ±10%, xoay ±15° | 0.5 |
| 7 | Affine (Shear) | Cắt xiên theo trục X/Y trong khoảng ±10°, | 0.3 |
| 8 | Perspective | Biến dạng phối cảnh ngẫu nhiên 3–10%, giữ nguyên kích thước ảnh | 0.5 |
| 9 | Mosaic | Kết hợp 4 ảnh ngẫu nhiên thành 1 ảnh lớn để tăng đa dạng spatial layout | 1.0 |
| 10 | CopyPaste | Sao chép đối tượng từ ảnh này sang ảnh khác (chỉ áp dụng cho segmentation) | 0.1 |
| 11 | MixUp | Trộn hai ảnh bằng cách lấy trung bình trọng số của pixel và nhãn | 0.1 |
| 12 | CutMix | Cắt một vùng ngẫu nhiên từ ảnh này và dán vào ảnh khác, điều chỉnh nhãn tương ứng | 0.0 |

# 4. Kiến trúc mô hình

## 4.1 YOLO11

YOLOv11 vẫn giữ cấu trúc ba phần Backbone–Neck–Head truyền thống nhưng đưa vào các module mới:

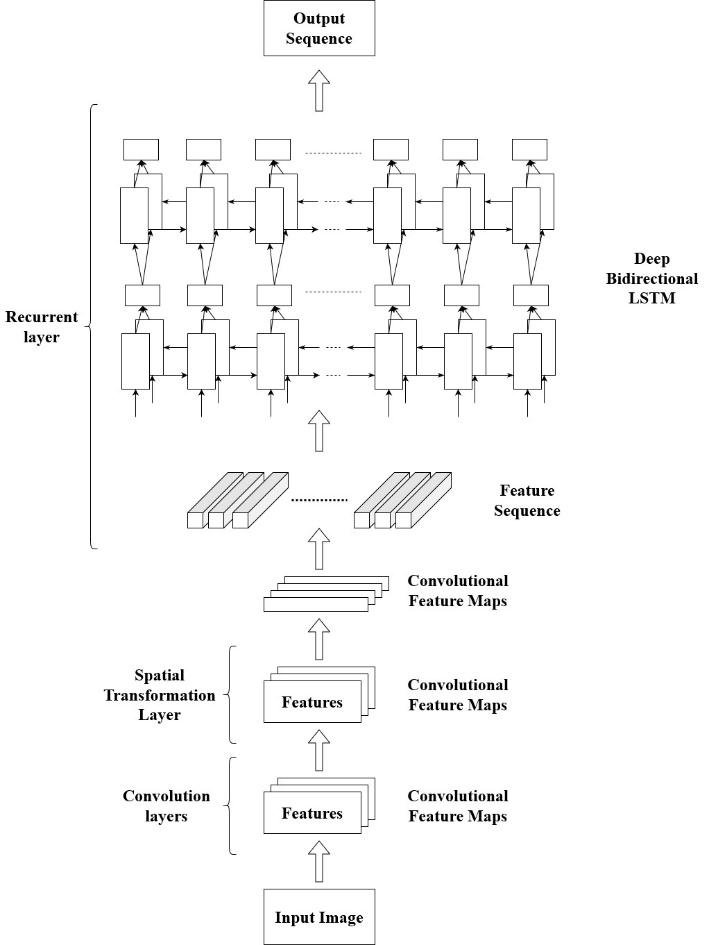
* **Backbone**: Chuỗi Convolution + C3K2 blocks để trích xuất đặc trưng.
* **Neck**: Kết hợp đa quy mô (feature fusion) qua các khối Upsample, Concat, SPFF và C2PSA để làm giàu thông tin không gian.
* **Head**: Ba nhánh Detect ở các kích thước khác nhau, mỗi nhánh gồm Convolution và Depth-wise Separable Convolution (DSC) để dự đoán bounding box và class.



| **Thành phần** | **Đầu vào → Đầu ra** | **Chức năng chính** |
| --- | --- | --- |
| **Backbone** | 640×640 → feature maps 20/40/80 | Conv→C3K2 (+SPPF,C2PSA) trích xuất đa độ phân giải |
| **Neck** | maps 20→40→80 | Upsample+Concat+C3K2 fusion đa quy mô |
| **Head** | maps → boxes, classes, masks | Conv subnet cho detection & mask |

## 4.2 CRNN

CRNN (Convolutional Recurrent Neural Network) – một mô hình “end-to-end” kết hợp CNN và RNN, thường dùng cho bài toán nhận dạng chuỗi từ hình ảnh (như OCR).



| **Thành phần** | **Đầu vào → Đầu ra** | **Chức năng chính** |
| --- | --- | --- |
| **CNN** | Image → feature maps | Conv→BN→ReLU→Pool trích xuất đặc trưng |
| **Bi-RNN** | feature maps → sequences | BiLSTM/GRU nắm bắt ngữ cảnh hai chiều |
| **CTC Layer** | RNN outputs → text | Chuyển per-frame scores thành chuỗi ký tự |
| **Loss (CTC Loss)** | preds, labels → loss | Tính loss kèm blank & alignment tự động |

# 5.Huấn luyện mô hình

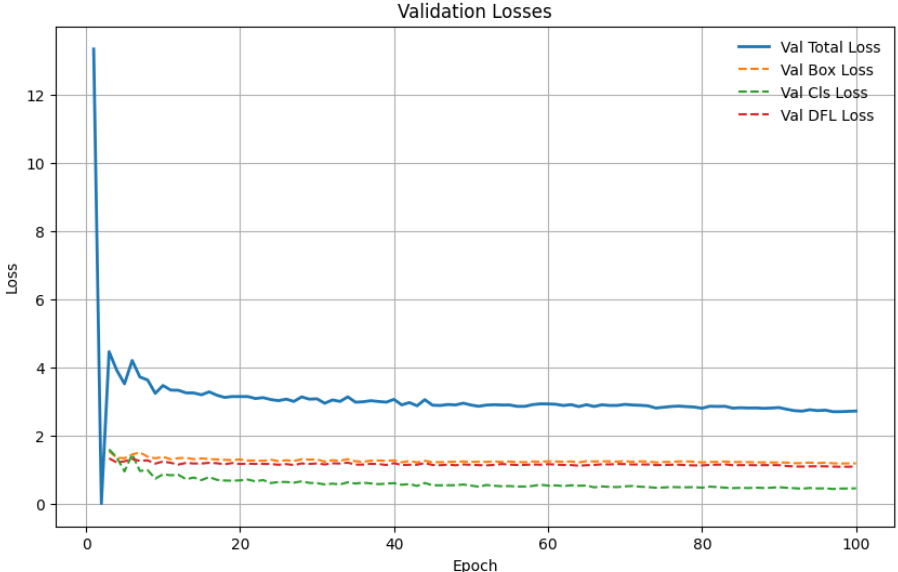
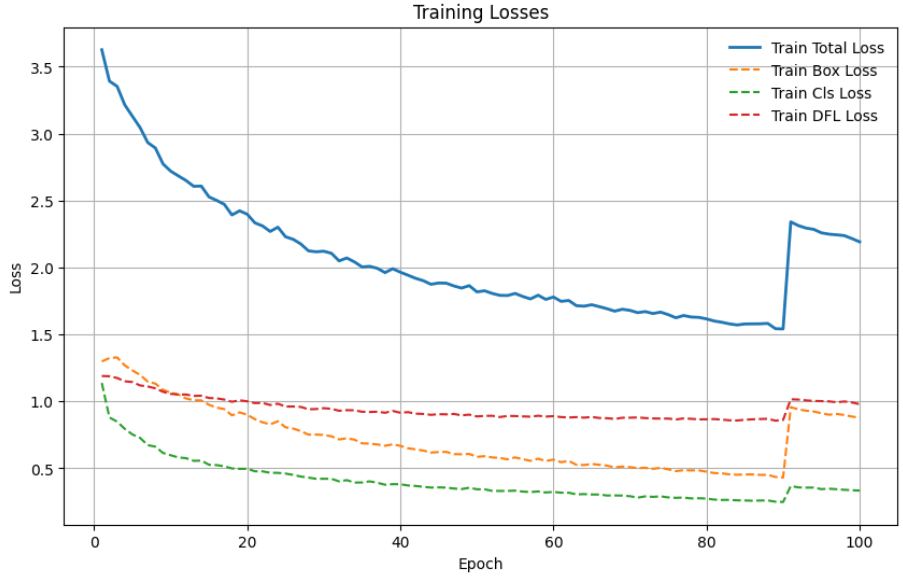
Yêu cầu:

Trình bày việc tinh chỉnh các siêu tham số, thông số của mô hình,…

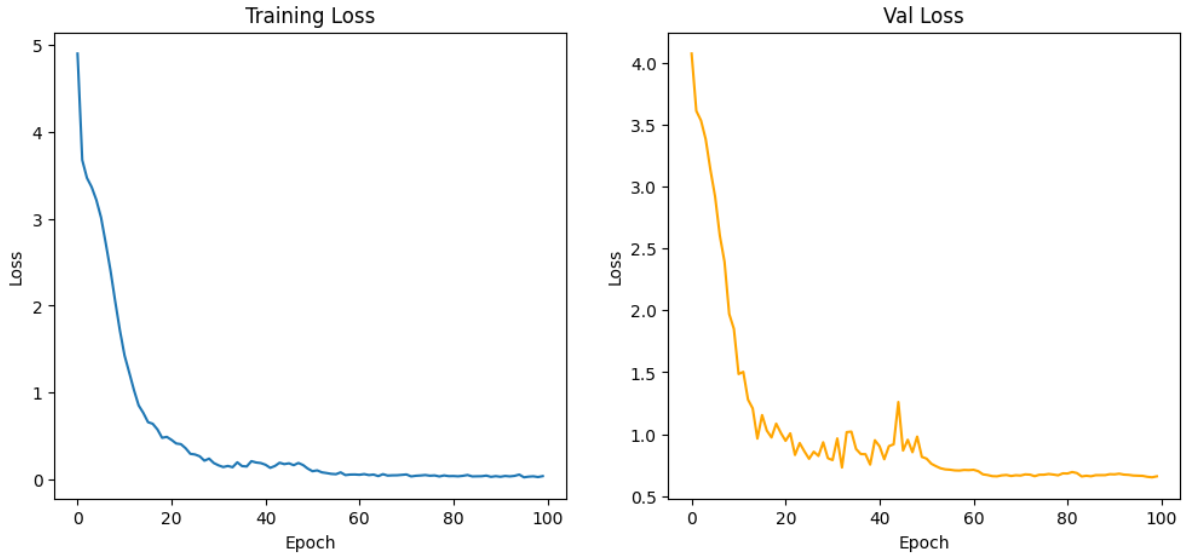
Các số liệu thu được trong quá trình huấn luyện (nên trình bày dạng sơ đồ, bảng biểu)

**YoloV11**

Tham số sử dụng: batch\_size = 16, lr = 0.002, optimizer = AdamW, epochs = 100



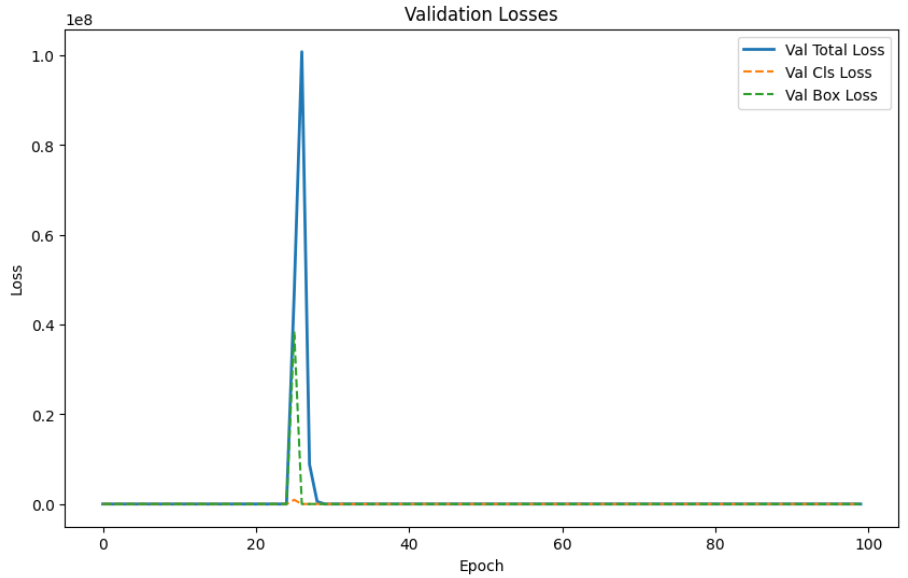
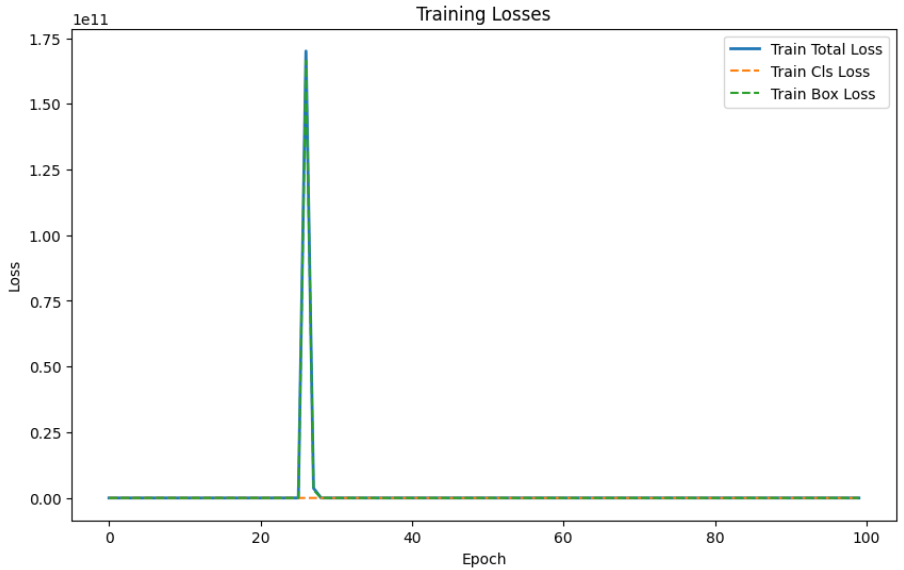
**CRNN:** Tham số sử dụng: batch\_size = 64, lr = 1e-3, optimizer = Adam, weight decay = 1e-5, epochs = 100, scheduler\_step\_size = epochs \* 0.5



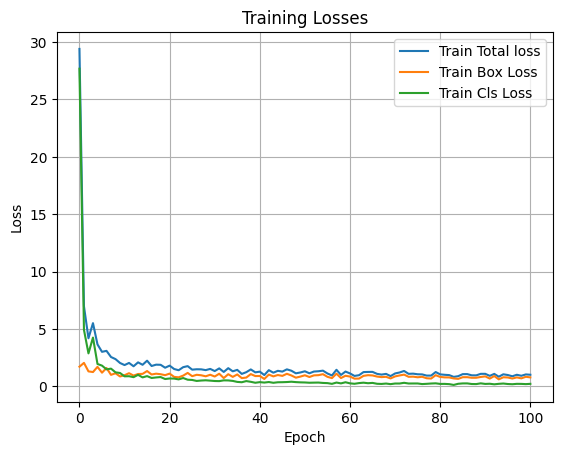
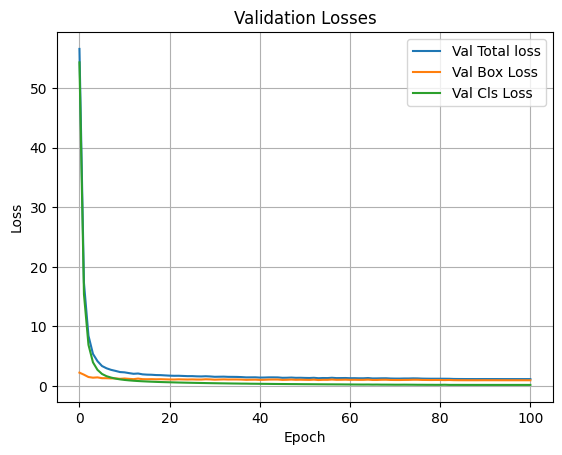
# 6. Kết quả thử nghiệm, so sánh, đánh giá

Kết quả so sánh với các model detection khác

**FasterRCNN** Tham số sử dụng: batch\_size = 15, lr = 0.001667, optimizer = AdamW, epochs = 100



**EfficientDet (b0):** Tham số sử dụng: batch\_size = 32, lr = 1e-3 , optimizer = AdamW, epochs = 100

# Kết quả đánh giá mô hình CRNN trên tập test ICDAR2003

|  |  |
| --- | --- |
| Chỉ số đánh giá | Giá trị (%) |
| CER | 9.09 |
| WER | 38.00 |

# 7. Kết luận

Dự án đã xây dựng thành công hệ thống Nhận dạng Văn bản trong Ảnh (STR) theopipelinehai giai đoạn, sử dụng **YOLOv11** cho phát hiện văn bản và **CRNN kết hợp CTC Loss** cho nhận dạng. Hệ thống được huấn luyện trên tập **ICDAR2003**, với quy trình xử lý và tiền xử lý dữ liệu được thực hiện bài bản.

### ****Ưu điểm:****

* **YOLOv11** có **tốc độ suy luận nhanh (~25ms/ảnh)**, phù hợp cho các ứng dụng thời gian thực.
* **Mô hình nhẹ (~12M tham số)**, dễ triển khai trên thiết bị biên.
* **CRNN** nhận dạng văn bản ổn định, cho kết quả đánh giá tốt:
  + **CER = 9.09%**, phản ánh độ chính xác ký tự cao.
  + **WER = 38.00%**, mức sai số hợp lý khi đánh giá trên văn bản chưa chuẩn hóa (giữ nguyên viết hoa, dấu câu, v.v.).
* Các kỹ thuật **data augmentation** giúp mô hình tổng quát hóa tốt, giảm thiểu hiện tượng bias vị trí.

### ****Nhược điểm:****

* Mặc dù loss của YOLOv11 đã ổn định sau ~20 epochs, nhưng **Train Loss vẫn quanh mức 2.2**, chưa đạt mức tối ưu lý tưởng.
* **WER còn cao (38%)**, cho thấy hệ thống còn hạn chế khi xử lý văn bản dài, chứa nhiều từ hoặc biến thể ngôn ngữ phức tạp.

### ****Tổng kết:****

Hệ thống STR đạt hiệu quả khá tốt trong bối cảnh bài toán đơn ngữ và có thể mở rộng trong tương lai với các mô hình mạnh hơn hoặc dữ liệu đa dạng hơn. Hướng phát triển tiếp theo có thể bao gồm fine-tuning CRNN với từ điển chuyên ngành và cải thiện pre-processing để giảm lỗi từ-level.

=== The End ===