# Deployment Guide: YOLOv8 Number Plate Detection Model

## 1. Local System Deployment

Requirements:

• Python 3.8+

• pip installed

• Trained model file (best.pt)

• GPU optional (CPU also works)

Step 1: Install Dependencies

* pip install ultralytics ,opencv-python ,easyocr, numpy

Step 2: Create a script named run\_local.py:

from ultralytics import YOLO  
import cv2  
  
model = YOLO("best.pt")  
img = cv2.imread("test.jpg")  
results = model.predict(img, conf=0.5)  
  
for r in results:  
 for box in r.boxes:  
 x1, y1, x2, y2 = box.xyxy[0].int().tolist()  
 cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)  
  
cv2.imshow("Result", img)  
cv2.waitKey(0)

Step 3: Run the script

* python run\_local.py

## 2. Deployment Using Streamlit

Install Streamlit:

* pip install streamlit

Create app.py:

import streamlit as st  
from ultralytics import YOLO  
import cv2  
import numpy as np  
  
model = YOLO("best.pt")  
st.title("Automatic Number Plate Recognition")  
  
uploaded = st.file\_uploader("Upload Image", type=["jpg","png"])  
  
if uploaded:  
 file\_bytes = np.frombuffer(uploaded.read(), np.uint8)  
 img = cv2.imdecode(file\_bytes, 1)  
 results = model.predict(img)  
  
 for r in results:  
 for box in r.boxes:  
 x1,y1,x2,y2 = box.xyxy[0].int().tolist()  
 cv2.rectangle(img,(x1,y1),(x2,y2),(0,255,0),2)  
  
 st.image(img, channels="BGR")

Run the app:

* streamlit run app.py

## 3. Cloud Deployment

Google Colab Deployment:

• Upload best.pt to Drive

• Install ultralytics and run script in notebook

AWS / Azure / GCP Deployment:

sudo apt update  
sudo apt install python3-pip -y  
pip install ultralytics opencv-python streamlit easyocr

Run: streamlit run app.py --server.port 80

## 4. Docker Deployment

Create Dockerfile:

FROM python:3.10  
WORKDIR /app  
COPY . /app  
RUN pip install ultralytics opencv-python streamlit easyocr  
EXPOSE 8501  
CMD ["streamlit", "run", "app.py", "--server.port=8501", "--server.address=0.0.0.0"]

Build & Run:

* docker build -t anpr .
* docker run -p 8501:8501 anpr

## 5. Recommended for Production

• GPU-enabled cloud server

• Docker containerization

• Nginx reverse proxy • HTTPS + Auto-restart setup