Mid Term Task: Integrasi YOLO dan Transformer untuk Analisis Perilaku Manusia pada Video Waktu-Nyata

Nadira Putri Wijaya

November 2, 2024

Abstract

Laporan ini bertujuan untuk mendokumentasikan hasil integrasi model YOLO (You Only Look Once) dengan Transformer dalam melakukan analisis perilaku manusia dalam video waktu-nyata. Laporan ini mencakup tinjauan mendalam mengenai teori algoritma deep learning yang digunakan, tantangan teknis, metode implementasi, optimasi model, serta hasil eksperimen dan analisis kinerja. YOLO digunakan untuk mendeteksi manusia dalam tiap frame video, sedangkan Transformer dimanfaatkan untuk menganalisis urutan temporal hasil deteksi dan mengidentifikasi pola perilaku manusia. Model ini berpotensi untuk diterapkan dalam aplikasi pengawasan keamanan dan pemantauan perilaku.

Contents

Chapter 1

Pendahuluan

1.1 Latar Belakang

Deteksi dan analisis perilaku manusia dalam video waktu-nyata memainkan peran penting dalam berbagai aplikasi, termasuk pemantauan keamanan, analisis lalu lintas pejalan kaki, serta pengawasan publik. Dengan menggunakan model YOLO untuk deteksi objek yang cepat dan Transformer untuk analisis pola temporal, sistem ini dapat memberikan informasi yang lebih rinci mengenai perilaku dan pergerakan manusia secara waktu-nyata.

1.2 Tujuan

Tujuan utama dari laporan ini adalah untuk:

- Menganalisis tantangan teknis dalam mengintegrasikan YOLO dengan Transformer untuk analisis perilaku manusia.
- Mengevaluasi keefektifan Transformer dalam menganalisis data urutan temporal dari hasil deteksi YOLO.
- Mengeksplorasi berbagai metode optimasi untuk meningkatkan performa waktu-nyata dari sistem yang diimplementasikan.

Chapter 2

Tinjauan Pustaka

2.1 YOLO (You Only Look Once)

YOLO adalah model deteksi objek berbasis CNN (Convolutional Neural Network) yang dirancang untuk kecepatan dan efisiensi dalam mendeteksi berbagai objek dalam suatu gambar atau video. Pada dasarnya, YOLO membagi gambar menjadi grid, di mana setiap sel grid bertanggung jawab untuk mendeteksi objek tertentu. YOLO memproses seluruh gambar sekali waktu, sehingga cocok untuk aplikasi real-time. YOLOv5 adalah versi yang banyak digunakan saat ini karena arsitekturnya yang dioptimalkan untuk GPU dan kemudahan integrasi dengan perangkat keras.

2.2 Transformer

Transformer adalah arsitektur deep learning berbasis self-attention yang unggul dalam analisis data sekuensial. Self-attention memungkinkan model untuk menangkap hubungan jangka panjang dalam data, sehingga cocok untuk analisis pola perilaku manusia dalam urutan frame video. Dibandingkan dengan model RNN, Transformer memiliki kemampuan yang lebih tinggi dalam mengidentifikasi pola kompleks dalam data waktu.

2.3 Deteksi Perilaku Manusia dalam Video Waktu-Nyata

Deteksi perilaku manusia dalam video memiliki tantangan unik, seperti kecepatan perubahan posisi objek, skala objek yang bervariasi, dan interaksi antar objek. Dengan YOLO, manusia dapat dideteksi sebagai objek spesifik di setiap frame, kemudian dianalisis oleh Transformer untuk mendapatkan pola perilaku yang mungkin terlewat oleh metode tradisional.

2.4 Algoritma Deep Learning yang Relevan

2.4.1 Convolutional Neural Networks (CNN)

CNN adalah salah satu arsitektur yang paling populer dalam deteksi objek. CNN berfokus pada ekstraksi fitur spasial dari data gambar melalui konvolusi, pooling, dan operasi aktivasi. Setiap lapisan konvolusi mengidentifikasi pola yang semakin kompleks pada data input, memungkinkan model mengenali objek dengan berbagai variasi bentuk dan posisi.

2.4.2 Recurrent Neural Networks (RNN)

RNN adalah model deep learning yang dirancang untuk data sekuensial. Meskipun populer dalam analisis urutan waktu, RNN memiliki keterbatasan dalam menangkap hubungan jangka panjang. Oleh karena itu, Transformer yang menggunakan self-attention mulai menggantikan RNN untuk tugas-tugas yang membutuhkan pemahaman hubungan urutan jangka panjang.

2.4.3 Self-Attention pada Transformer

Self-attention dalam Transformer memungkinkan model untuk memberikan bobot yang berbeda pada setiap bagian urutan, tergantung pada hubungan dan kontribusinya terhadap hasil akhir. Dalam konteks analisis perilaku manusia, self-attention dapat membantu model mengidentifikasi perubahan perilaku yang mungkin terkait dengan aktivitas tertentu, seperti berjalan, berlari, atau gerakan mencurigakan lainnya.

Chapter 3

Implementasi di Google Colab

3.1 Instalasi dan Setup Library

Di Google Colab, kita menginstal library yang dibutuhkan untuk deteksi objek dengan YOLO dan analisis video dengan OpenCV. PyTorch digunakan untuk menjalankan YOLOv5, dan Transformer dari Hugging Face digunakan untuk analisis urutan:

```
!pip install torch torchvision torchaudio
!pip install transformers opencv-python-headless
!git clone https://github.com/ultralytics/yolov5.git
!pip install -qr yolov5/requirements.txt
```

3.2 Upload Video dan Persiapan YOLO untuk Deteksi Manusia

Karena Google Colab tidak mendukung input webcam langsung, video diunggah dari Google Drive. Model YOLOv5 kemudian diimpor dan disiapkan untuk mendeteksi manusia dalam setiap frame video:

```
from google.colab import drive
drive.mount('/content/drive')

import torch
model = torch.hub.load('ultralytics/yolov5', 'yolov5s', pretrained=True)
```

3.3 Deteksi Objek Manusia dengan YOLOv5

Model YOLOv5 mendeteksi manusia dalam tiap frame, dan hasil deteksi disimpan dalam bentuk bounding box dan label:

```
results = model(frame)
detections = results.pandas().xyxy[0]
detections = detections[detections['name'] == 'person']  # Filter only human det
```

3.4 Preprosesing dan Penyimpanan Hasil Deteksi untuk Analisis Transformer

Untuk menyiapkan input yang sesuai bagi Transformer, hasil deteksi dinormalisasi dan disimpan dalam format yang terstruktur, mempertahankan identitas objek antar frame.

```
normalized_data = []
for _, row in detections.iterrows():
    bbox = [row['xmin'], row['ymin'], row['xmax'], row['ymax']]
    normalized_data.append(bbox)
```

3.5 Pelacakan Identitas Manusia Antar Frame

Dengan teknik pelacakan objek, kita dapat mempertahankan identitas manusia pada setiap frame video. Teknik sederhana ini menggunakan perbandingan posisi antara frame sebelumnya dan saat ini untuk mempertahankan konsistensi identitas objek.

3.6 Optimasi Model untuk Performa Waktu-Nyata

3.6.1 Model Quantization

Quantization digunakan untuk mengurangi ukuran model dan mempercepat inferensi:

```
model = torch.quantization.quantize_dynamic(model, dtype=torch.qint8)
```

3.6.2 Pengaturan Resolusi Input

Mengurangi resolusi input memungkinkan YOLO untuk memproses frame lebih cepat, meskipun ada sedikit kompromi pada akurasi deteksi.

```
frame = cv2.resize(frame, (640, 360))
```

Chapter 4

Hasil dan Analisis

4.1 Evaluasi Keefektifan Transformer untuk Analisis Perilaku

Model Transformer dapat mengidentifikasi pola perilaku manusia dari urutan hasil deteksi YOLO. Dengan self-attention, Transformer dapat menganalisis perubahan posisi manusia di berbagai frame, memungkinkan deteksi pola perilaku seperti pergerakan tiba-tiba yang mencurigakan.

4.2 Pengaruh Optimasi pada Performa Waktu-Nyata

Setelah optimasi, waktu pemrosesan berkurang dengan kompromi akurasi yang minimal. Pada eksperimen ini, pengurangan resolusi input sedikit mengurangi akurasi deteksi, tetapi memberikan hasil yang cukup akurat untuk aplikasi waktu-nyata.

4.3 Studi Kasus: Deteksi Perilaku Mencurigakan

Dengan mengkombinasikan YOLO dan Transformer, kita dapat mendeteksi perilaku mencurigakan, seperti pergerakan mendadak atau aktivitas abnormal di area tertentu. Studi ini menunjukkan bahwa self-attention pada Transformer dapat memperkuat deteksi aktivitas yang memerlukan perhatian khusus dalam pemantauan keamanan.

Chapter 5

Kesimpulan

Integrasi YOLO dan Transformer untuk deteksi dan analisis perilaku manusia secara waktu-nyata menunjukkan potensi besar dalam aplikasi keamanan. Optimasi model sangat penting untuk mencapai performa yang memadai dalam kondisi terbatas, seperti pada perangkat mobile. Laporan ini memberikan tinjauan menyeluruh terhadap tantangan teknis dan hasil yang dicapai dalam tugas ini.