Violence Detection in Video

Introduction

The prevalence of violent incidents captured on video has significantly increased with the widespread use of surveillance systems, smartphones, and social media platforms. Detecting violence in videos has become a critical necessity for enhancing public safety, preventing crimes, and enabling rapid response in emergencies. Automated violence detection systems can assist law enforcement agencies, private security firms, and public organizations in monitoring real-time threats, reducing human monitoring fatigue, and enabling proactive interventions.

The use cases for such systems are vast, ranging from surveillance in public spaces like malls, schools, and transport hubs to content moderation on online platforms to prevent the spread of harmful content. Additionally, violence detection systems can be employed in forensic video analysis and assist in maintaining the safety of staff and inmates in correctional facilities.

Traditional methods for violence detection relied heavily on manual monitoring or heuristic-based approaches, which are not only time-intensive but also prone to inconsistencies and inaccuracies. With advancements in computer vision and deep learning, automated systems can now achieve significant accuracy in analyzing video streams. Methods leveraging convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transfer learning have demonstrated the potential to identify violent actions with high precision by analyzing motion patterns, object interactions, and scene contexts.

This project explores the implementation of a deep learning-based approach to detect violence in videos. By combining state-of-the-art models and video preprocessing techniques, the system aims to achieve reliable and efficient violence detection, addressing critical gaps in existing methods while highlighting the potential for real-world applications.

Dataset

The dataset used for this project is taken from Kaggle: "Real Life Violence Situations Dataset". It contains 1000 violent videos and 1000 non-violent videos. The videos are of varying lengths and frame sizes, containing a variety of violent and non-violent situations. The violent situations occur in various settings ranging from public brawls to violent sports, and the non-violent situations occur in normal daily life, sports, dance and music.
Some videos had problems in compression or number of frames which made them unsuitable for use and had to be dropped from the dataset. After which the videos were split into train and test sets in an 80:20 ratio.

Individual work by Dhanush Bhargav

1. Creating the data loader for loading videos and building train and test pipelines
2. Developing and training ResNet3D model for video classification
3. Creating functionality to run inference on individual video files.
4. Developing streamlit application to upload video files and obtain inference

Detailed description of work

Data Loader

1. Wrote code to create train and test CSV files containing video file names and labels (1=violent, 0=nonviolent).
2. Wrote dataset class for reading, transforming and returning video frames and labels.

ResNet3D model

1. Used pretrained ResNet3D model for video classification as a backbone to develop the model for violent video classification.
2. Wrote a function to train the model.
3. Wrote a function to test model on the test data and obtain performance metrics.

Inference Functionality

1. Developed a class to load the trained model and run inference on a single video.
2. Developed a function to extract clips from a single video.
3. Exposed this functionality to the streamlit application.

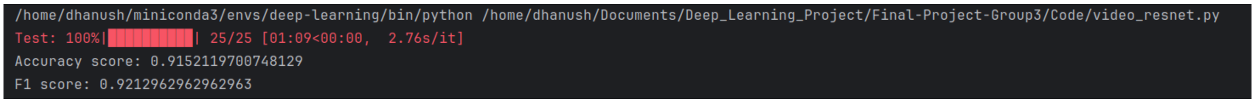
Streamlit Application

1. Contributed towards developing the Streamlit application which allows users to upload a video file and then runs inference on the file and displays the annotated video.

Result

Video Classifier : R3D_18

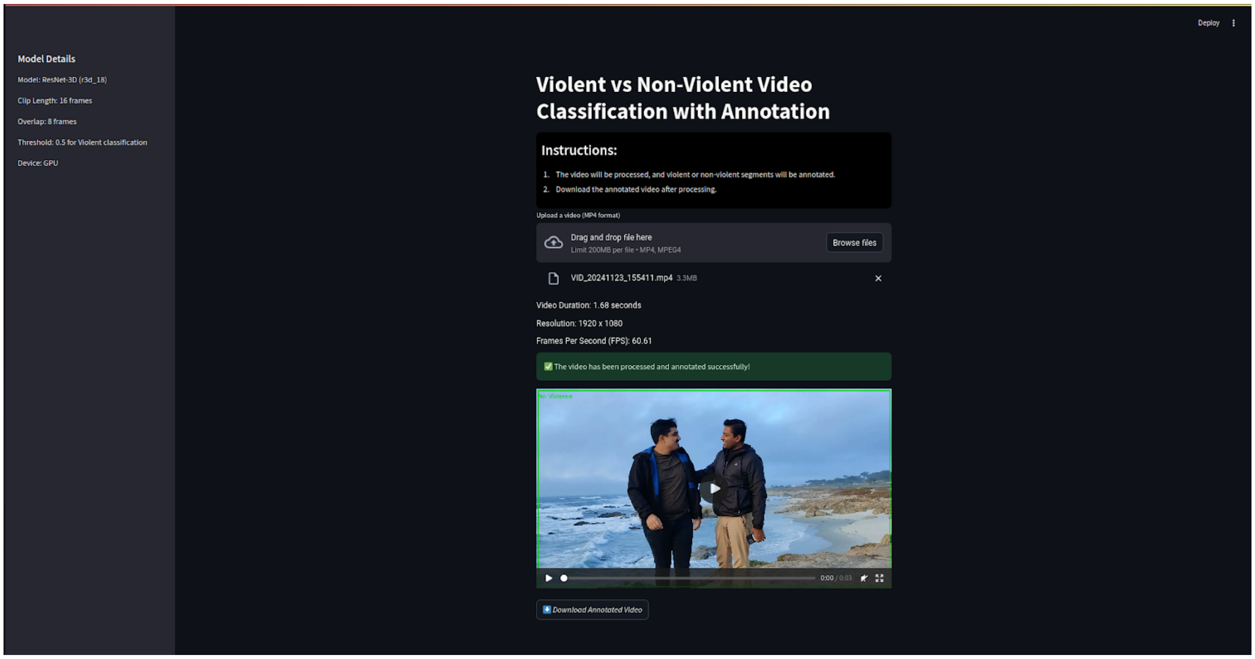
The below image shows the performance metrics of the fine-tuned R3D_18 video classification model on the test set.



```
/home/dhanush/miniconda3/envs/deep-learning/bin/python /home/dhanush/Documents/Deep_Learning_Project/Final-Project-Group3/Code/video_resnet.py
Test: 100%|██████████| 25/25 [01:09<00:00,  2.76s/it]
Accuracy score: 0.9152119700748129
F1 score: 0.9212962962962963
```

Streamlit Application

Below is a screenshot of the streamlit application with an annotated result video.



Model Details

Model: ResNet-3D (r3d_18)

Clip Length: 16 frames

Overlap: 8 frames

Threshold: 0.5 for Violent classification

Device: GPU

Violent vs Non-Violent Video Classification with Annotation

Instructions:

1. The video will be processed, and violent or non-violent segments will be annotated.
2. Download the annotated video after processing.

Upload a video (MP4 format)

Drag and drop file here
Limit 200MB per file • MP4, MPEG4

VID_20241123_155411.mp4 3.3MB

Video Duration: 1.68 seconds

Resolution: 1920 x 1080

Frames Per Second (FPS): 60.61

✅ The video has been processed and annotated successfully!

Summary

Through my contributions in this project, I learnt important skills in data preparation, model finetuning, and maintaining a codebase for a project. I also learnt how to work collaboratively as part of a team of peers.

Future improvements I would like to make to this project, is making a real-time application that uses a live video feed to identify and flag violent situations in real-time.

Code percentage

Lines taken from internet = 80
Lines modified and used from internet = 40
Lines written by self = 150

Percentage of lines copied from internet = (80 - 40) / (80 + 150) x 100 = 17.391%
Percentage of lines taken from the internet = **17.40%**

References

1. https://pytorch.org/vision/main/models/generated/torchvision.models.video.r3d_18.html#torchvision.models.video.r3d_18
2. https://pytorch.org/tutorials/beginner/basics/data_tutorial.html