Case Study 2

Security Surveillance Anomaly Detection

Task 1: Develop Anomaly Detection Algorithm for Surveillance Systems

1. Introduction

The objective of this task is to develop a robust anomaly detection algorithm for surveillance systems. The algorithm identifies unusual events in real-time video streams, facilitating proactive monitoring and response in security and safety applications.

2. Methodology

Model Development

Model Choice:

- **CNN Model:**
  - **Architecture:** Convolutional Neural Network (CNN)
  - **Details:** Input shape of (50, 50, 1), multiple Conv2D layers with LeakyReLU activation, MaxPooling2D layers, Dropout layers for regularization, and Dense layers for classification.
  - **Output:** Flattened and connected to a final Dense layer with softmax activation.
- **LSTM Model:**
  - **Architecture:** Long Short-Term Memory (LSTM) Network
  - **Details:** Input shape of (2500, 1), two LSTM layers with tanh activation, Dense layer, and Dropout for regularization.
  - **Output:** Flattened for further processing.
- **Combined Model:**
  - Combines CNN and LSTM models using Keras' concatenate layer.
  - Output connected to a final Dense layer with softmax activation for classification.

Training Process:

Data Source:

- Utilized the UCF Crime Dataset, comprising 64x64 grayscale images from surveillance videos across 14 classes.

Data Preparation:

- Preprocessed using OpenCV, resizing to 50x50 pixels, and converted into a 4D array for Keras' ImageDataGenerator.
- Split into training and test sets (90:10 ratio) for CNN and LSTM models.

- Labels encoded categorically for alignment with model output.

Model Architecture:

- CNN with three convolutional layers (64, 128, 256 filters), LeakyReLU activation, max-pooling, and dropout.
- LSTM with two layers (8 units each), tanh activation.
- Combined output through concatenate layer, final softmax layer for anomaly classification.
- Optimized with Adam, trained over 20 epochs, monitored via ModelCheckpoint and CSVLogger.

Training Objective:

- Optimized CNN-LSTM model to detect anomalies effectively in UCF Crime Dataset videos.
- Spatial features learned using CNN layers, temporal dependencies with LSTM layers.

3. Evaluation

- Achieved 99.68% accuracy on test data, with precision, recall, and F1-scores exceeding 99% across seven anomaly types.
- Confusion matrix shows minimal misclassifications, IoU score of 99.37% confirms precise anomaly detection.
- Low training and test losses (0.60% and 0.87%, respectively) demonstrate model efficiency.

Real-Time Anomaly Detection Using CNN-LSTM Model in Streamlit

Overview:

- Developed Streamlit app for real-time anomaly detection.
- Utilizes CNN-LSTM model trained on UCF Crime Dataset.

Features:

- Video Upload: Supports MP4 or AVI formats.
- Real-Time Processing: Sequential frame analysis on user request.
- Frame-by-Frame Analysis: CNN-LSTM model applied to each frame.
- Anomaly Detection: Frames exceeding 0.5 anomaly probability flagged.
- Anomaly Classification: Categories include fighting, shoplifting, abuse, arrest, shooting, robbery, or explosion.
- Probability Display: Displays anomaly probability per frame.

Functionality:

- Frame Display: Streams annotated frames indicating anomaly presence.
- Interactive Playback: User-controlled playback speed and frame navigation.
- Results Visualization: Highlights frames with detected anomalies, type, and probability.

Task 2: Test the solution using historical data and fine-tune parameters for accuracy.

In task 2, our CNN-LSTM model was tested on CCTV footage depicting abuse and robbery instances. The model processed video frames in real-time, accurately detecting and classifying anomalies. The application promptly notified anomaly presence, confirming its effectiveness in real-world surveillance.

Task 3: Recommendations for Response Protocols

Based on detected anomalies, recommendations include:

1. Immediate Alert Mechanism: Real-time alerts via alarms or notifications.
2. Priority Response Allocation: Prioritize based on anomaly severity.
3. Enhanced Surveillance Focus: Optimize camera placement and coverage.
4. Regular Training and Drills: Ensure preparedness of security personnel.
5. Continuous System Evaluation: Update model and evaluate performance.
6. Legal Compliance: Adhere to privacy and regulatory guidelines.

Conclusion

The CNN-LSTM anomaly detection system developed using the UCF Crime Dataset excels in identifying anomalies like fighting, shoplifting, abuse, arrest, shooting, robbery, and explosion. It combines robust deep learning models with real-time processing capabilities via a Streamlit application. Achieving high accuracy and minimal misclassifications, the system offers reliable surveillance enhancement. Future improvements will focus on fine-tuning parameters and expanding surveillance capabilities while adhering to ethical and legal guidelines.