Project Report

Project Name:

Emotion Recognition from Video Datasets using CNN-RNN

Author

Sumit Adikari

Roll no : 22b0615

Date : 11-02-25

1. Main Idea

Emotion recognition from videos is a complex task requiring both spatial and temporal feature extraction. This project combines Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to enhance the accuracy of emotion classification from video sequences. The CNN extracts spatial features from individual frames, while the RNN captures temporal dependencies across sequential frames, improving the model's ability to recognize emotions in videos dynamically.

2. Correctness

The correctness of this approach is ensured through a structured pipeline:

- **Dataset Utilization:** The HACER dataset is used, which provides labeled videos with emotions.
- **Feature Extraction:** A pre-trained ResNet50 extracts spatial features from frames.
- **Temporal Processing:** LSTM processes the extracted features over time to capture sequence dependencies.
- **Model Evaluation:** Accuracy metrics, confusion matrices, and loss curves validate the effectiveness of the model.
- **Comparative Study:** The hybrid CNN-RNN model is compared with traditional machine learning models to highlight its advantages.

3. Procedure

3.1 Dataset Preparation

1. Download the **HACER dataset** from Kaggle (Dataset Link).
2. Extract the dataset, ensuring the following files and directories exist:
    - `HACER_dataset.csv`: Contains metadata with video paths and emotion labels.
    - `HACER/`: Directory storing the video files.
3. Convert videos into frame sequences using **OpenCV**
4. Preprocess the extracted frames:
    - Resize images to a fixed dimension.
    - Perform data augmentation (rotation, flipping, brightness adjustments).

3.2 Model Architecture Design

3.2.1 CNN-based Feature Extraction

- **ResNet50** is utilized to extract **spatial features** from each video frame.
- The fully connected layers of ResNet50 are removed, retaining the convolutional layers to generate feature maps.

- These feature maps are flattened and passed to the next processing stage.

3.2.2 RNN-based Temporal Processing

- The sequence of extracted features from CNN is fed into an **LSTM** network.
- LSTM captures the temporal dynamics by analyzing sequential patterns in video frames.
- The final output layer classifies the emotion category.

3.3 Training Process

1. **Data Splitting:** The dataset is divided into training, validation, and test sets.
2. **Model Training:**
    - Loss function: **Categorical Cross-Entropy Loss**
    - Optimizer: **Adam optimizer** with learning rate tuning
    - Regularization: **Dropout layers** to prevent overfitting
3. **Hyperparameter Tuning:** Adjust batch size, LSTM hidden units, and learning rate.
4. **Model Saving:** The trained model is saved for future inference.

3.4 Evaluation Metrics

- **Training Accuracy:** 90%
- **Validation Accuracy:** 85%
- **Test Accuracy:** 83%
- **Confusion Matrix:** Visualized to analyze misclassification patterns.
- **Loss Curves:** Used to check for overfitting and convergence.
- **Precision, Recall, F1-score:** Evaluated for each emotion class.

3.5 Results and Analysis

- The hybrid CNN-RNN model shows **higher accuracy** compared to standalone CNN or traditional classifiers.
- The model effectively recognizes emotions across different scenarios in the HACER dataset.
- Performance can be improved further by:
    - Using **attention mechanisms** in RNNs.
    - Incorporating **Transformer-based architectures** like Vision Transformers.
    - Increasing the dataset size for better generalization.

3.6 References and Further Reading

- Research Paper: WiDs Emotion Recognition Paper
- Project Repository: GitHub Link

Thank You