Evaluation of Vital Sign Extraction from Video Feed Using Text Detection and Recognition Models

1. Introduction

The report outlines the process of developing a text detection and recognition-based pipeline for extracting vital signs from a video feed. The key objective is to develop an automated system capable of processing video frames, detecting relevant text, and extracting parameters such as ECG, SpO2, and BP using machine learning models.

2. System Overview

This project integrates several key components:

- **Text Detection Model**: A convolutional neural network (CNN)-based model designed to detect regions of interest (ROI) where text appears.
- **Text Recognition Model**: A sequence-based model (Bidirectional GRU with CNN layers) to recognize text within detected regions.
- **Vital Sign Extraction**: Parsing the recognized text using regular expressions to extract values for ECG, SpO2, and BP.
- **CSV Saving**: Storing the extracted data in CSV format for later analysis.

3. Methodology

3.1 Preprocessing of Video Frames

Each frame from the video feed is preprocessed as follows:

- Conversion to grayscale for uniform processing.
- Resizing to a standard size (128x128 pixels) to match the input requirements of the detection model.
- Normalization of pixel values to the range [0, 1].

3.2 Text Detection

- The model used for text detection is a CNN with three convolutional layers followed by max-pooling layers to extract features. The final output is a bounding box (x, y, w, h) for each detected region of text.

3.3 Text Recognition

- Once text regions are detected, the recognition model, which is a Bidirectional GRU-based architecture with convolutional layers, processes the detected regions.
- The recognized text is then parsed using regular expressions to extract vital signs.

3.4 Vital Sign Parsing

- **ECG**: Extracted using a regex pattern to match the format ECG: <value>.
- **SpO2**: Extracted using a regex pattern to match the format SpO2: <value>.
- **BP**: Extracted using a regex pattern to match the format BP: <systolic/diastolic>.

4. Performance Metrics

The performance of the system is evaluated using the following metrics:

- **Accuracy**: The percentage of correctly extracted vital signs compared to the ground truth.
- **Precision**: The ratio of correctly predicted positive observations to the total predicted positives.
- **Recall**: The ratio of correctly predicted positive observations to all observations in the actual class.
- **F1-Score**: The weighted average of precision and recall.
- **Error Rate**: The percentage of frames in which no vital signs were correctly detected.

Final Accuracy and Performance Results:

- Since the models have not been trained on labeled data, the actual performance metrics cannot be provided without running tests on a properly labeled dataset. Once trained, accuracy, precision, recall, and F1-score can be evaluated on a test dataset with known vital signs.

5. Observations

Strengths:

- The pipeline is modular, allowing easy replacement of components (e.g., models for detection and recognition).
- The text detection model is simple and can potentially be improved by incorporating pre-trained object detection models like YOLO or Faster R-CNN.
- The text recognition system, though basic, uses a GRU-based approach that can be fine-tuned for text-heavy applications like vital sign extraction.

Limitations:

- **Model Training**: The detection and recognition models require training on a sufficiently large and labeled dataset. Without training, the models won't perform accurately.
- **Limited Text Recognition**: The recognition model's performance will depend heavily on the quality and consistency of the text data in the video. Variations in font size, lighting conditions, and background noise may reduce the accuracy.
- **Bounding Box Detection**: If the bounding boxes generated by the text detection model are not accurate, the recognition model will struggle to identify the correct text.

6. Proposed Improvements

6.1 Improved Text Detection

- Incorporate pre-trained models like **YOLO** or **Faster R-CNN** for more accurate text region detection. These models are capable of detecting and classifying text regions with high precision.
- Use **Non-Maximum Suppression (NMS)** to eliminate redundant bounding boxes, improving the efficiency and accuracy of text region detection.

6.2 Enhanced Text Recognition

- Implement **CRNN** (Convolutional Recurrent Neural Networks), which is specifically designed for text recognition tasks and has shown superior performance on OCR-like tasks.
- Train the recognition model on a more diverse dataset containing various fonts, background noise, and distorted text.
- Consider leveraging pre-trained OCR models like **Tesseract OCR** for better text recognition, especially for simple vital sign readings.

6.3 Stable Video Feed Interpretation

- Use temporal information to stabilize text detection. Implementing **Optical Flow** or tracking algorithms could help maintain consistency when text moves across frames.
- Incorporate **contextual information**: Analyze the temporal sequence of frames to increase the confidence in detecting the same vital signs across multiple frames.
- Implement **multi-frame processing** to improve recognition accuracy by analyzing the text over multiple frames before making final extraction decisions.

6.4 Data Augmentation and Regularization:

- To improve model robustness, use data augmentation techniques such as random rotations, flips, and changes in lighting to make the model invariant to these changes.
- Use **dropout layers** in both the detection and recognition models to avoid overfitting, particularly when using small datasets.

7. Conclusion

This system provides a robust approach to extract vital signs from video frames through text detection and recognition. The modular nature allows for easy upgrades to each component, and the ability to process real-time video feeds is a promising feature. However, the performance heavily depends on the quality of training data and the chosen models.

In its current state, the system is more of a proof-of-concept and needs further enhancement, including model training, data collection, and integration of more advanced detection and recognition techniques. Once fully trained and optimized, this system could potentially be deployed in healthcare or medical monitoring systems for real-time vital sign extraction