BSL Digit Recognition Project

Project Overview

The BSL (British Sign Language) Digit Recognition project involves the creation of a machine learning system to accurately recognize digits signed in British Sign Language. This system uses computer vision techniques to process video input, extract hand landmarks, and classify the digits using a trained model. The project is structured into three main components: data collection and preprocessing, model building and training, and real-time digit recognition.

Components and Methodology

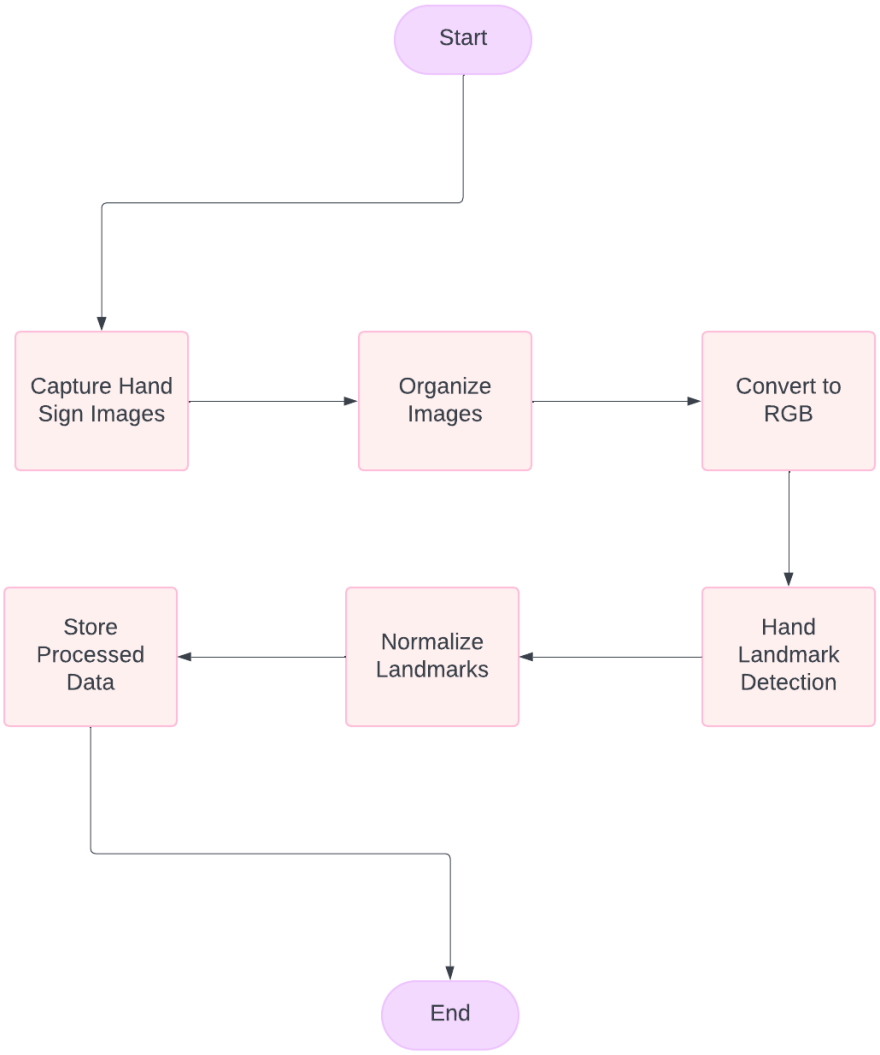
1. Data Collection and preprocessing (dataset builder)

- **Data Collection**:
- Captures images of hand signs using a webcam.
- Organizes images into directories corresponding to each digit class.
- Utilizes OpenCV to handle video capture and frame extraction.

- **Preprocessing**:
- Converts images to RGB.
- Uses MediaPipe to detect hand landmarks in each image.
- Extracts and normalizes landmark coordinates, storing them along with labels for training.

Summary:

- Creates a dataset of hand images for each digit.
- Uses MediaPipe to extract and normalize hand landmarks from images.
- Saves processed data for model training.



2. Model Building and Training (model builder)

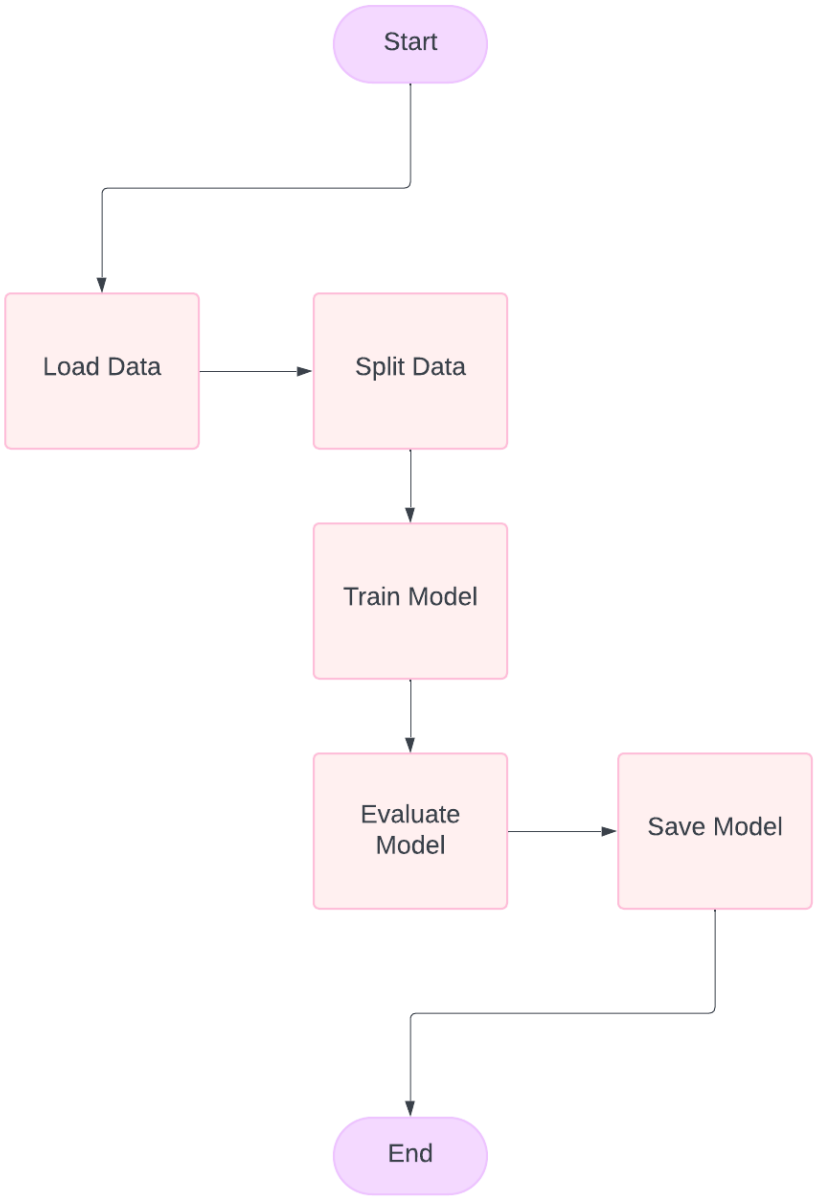
- **Loading Data**: Loads preprocessed hand landmark data and labels.

- **Model Training:**
- Splits the data into training and testing sets.
- Trains a Random Forest Classifier on the training data.
- Evaluates the model's performance on the test data.

- **Model Saving**: Saves the trained model to disk for later use.

Summary:

- Uses scikit-learn for data splitting, model training, and evaluation.
- Saves the trained model using pickle for easy deployment.



3. Real-Time Digit Recognition (main machine)

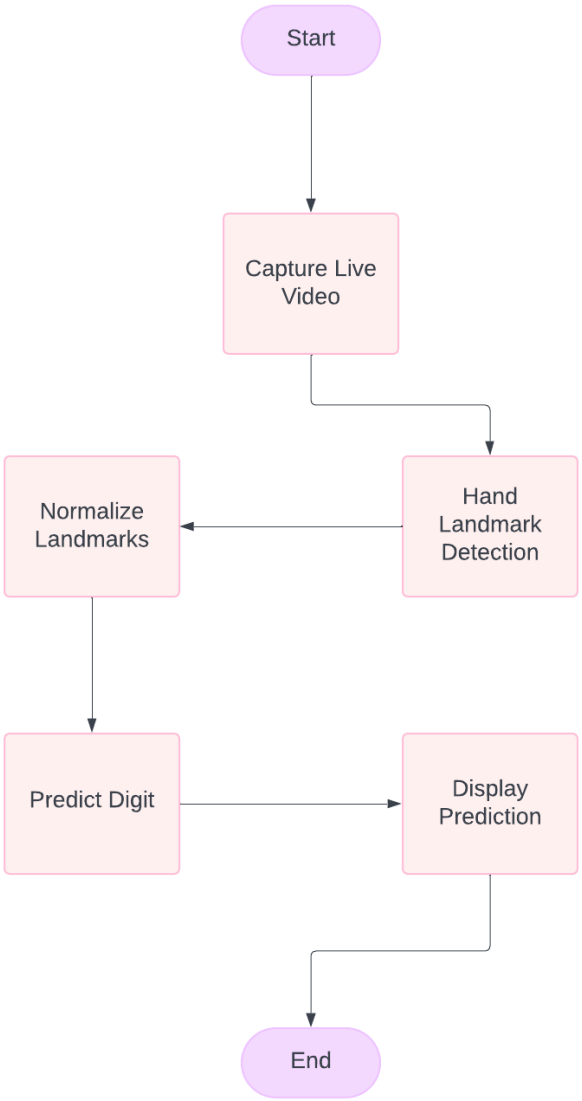
Real-Time Video Capture: Captures live video from the webcam.

- **Hand Landmark Detection**:
- Uses MediaPipe to detect hand landmarks in real-time.
- Normalizes the detected landmarks for model input.

- **Prediction and Display**:

- Uses the trained model to predict the digit being signed.
- Displays the predicted digit on the video feed with bounding boxes around the detected hand.

Summary:
- Integrates video capture with real-time hand landmark detection.
- Uses the trained model to make real-time predictions.
- Provides visual feedback by displaying predictions on the video feed.



Project Workflow

1. **Dataset Creation**: Run `dataset builder.py` to collect and preprocess images of hand signs, storing normalized hand landmarks and labels.

2. **Model Training**: Run `model builder.py` to train a Random Forest Classifier on the preprocessed data and save the trained model.

3. **Real-Time Recognition**: Run `main machine.py` to capture live video, detect hand landmarks in real-time, and use the trained model to recognize and display signed digits.

Outcomes

- A trained model capable of recognizing BSL digits with high accuracy.
- A real-time application that can accurately recognize and display signed digits from live video input.
- An organized and reusable codebase for further improvements and extensions, such as incorporating more sign language gestures or using more advanced deep learning models.

Applications

- **Educational Tools**: Develop interactive applications to teach and practice BSL.
- **Assistive Technology**: Create devices or software to aid communication for individuals who rely on sign language.
- **Research and Development**: Provide a foundation for further research in sign language recognition and related fields.

By leveraging computer vision and machine learning techniques, this project aims to create an effective and user-friendly system for recognizing and interpreting British Sign Language digits in real-time.