Video Summarization System

Backend Documentation

System Overview

This is an AI-powered video summarization system that provides multiple processing options:

Smart Edit: Intelligently trims videos while preserving the original speaker's voice
Narrated Summary: Creates AI-narrated video summaries with voice-over
Auto Captions: Generates automatic speech-to-text captions
Visual Summary: Creates keyframe-based timelapse videos

Backend Architecture

1. Main Backend Server

File: app.py (91 lines)
Framework: Flask 3.1.2
Port: 5000

Responsibilities:
- Handles HTTP GET/POST requests
- Manages file uploads to uploads/ directory
- Routes processing requests to appropriate modules
- Renders HTML templates with results
- Serves processed videos and static files

2. Processing Modules

Module	Purpose	Lines	Key Technology
auto_caption.py	Speech-to-text transcription	125	Whisper AI, FFmpeg
smart_edit.py	Intelligent video trimming	252	MoviePy, NLP
smart_cutter.py	AI narrated summaries	364	BART, Edge-TTS
video_summarizer.py	Keyframe extraction	160	OpenCV, FFmpeg
text_summarizer.py	Text summarization	5	BART model

AI Models Used

Model	Purpose	Size	Provider
Whisper-base	Speech recognition	140 MB	OpenAI
BART (facebook/bart-large-cnn)	Text summarization	800 MB	Meta AI
Edge-TTS (en-US-GuyNeural)	Text-to-speech	Cloud API	Microsoft

Request Processing Flow

Example: Smart Edit Request

1. **User Upload:** User uploads video via web interface

2. **Flask Handler:** app.py receives POST request, saves to uploads/input.mp4

3. **Smart Edit Processing:**
→ Calls smart_edit.create_smart_edit()
→ Step 1: Transcribe audio (Whisper AI)
→ Step 2: Analyze transcript, identify key segments
→ Step 3: Remove filler words (um, uh, like)
→ Step 4: Extract video clips WITH original audio
→ Step 5: Concatenate clips using MoviePy
→ Step 6: Write final MP4

4. **Return Results:** Send edited video + summary to frontend

5. **User Views:** Video player displays edited result

Backend Dependencies

Package	Version	Purpose
flask	3.1.2	Web framework
transformers	4.57.6	AI models (Whisper, BART)
torch	2.10.0	PyTorch backend for AI
moviepy	2.2.1	Video editing and processing
opencv-python	Latest	Computer vision, frame extraction
edge-tts	7.2.7	Microsoft Text-to-Speech
soundfile	Latest	Audio file I/O
mutagen	1.47.0	Audio metadata reading
numpy	Latest	Numerical computations

API Endpoints

GET /
Returns the main web interface (index.html)

POST /
Processes video with selected options
Parameters:
• video_file (file): The uploaded video
• do_smart_edit (checkbox): Enable smart editing
• do_narrated (checkbox): Enable AI narration
• do_caption (checkbox): Enable captions
• do_summary (checkbox): Enable visual summary

GET /uploads/<filename>
Serves processed video files
Returns: MP4 video file

GET /static/<path>
Serves CSS, JavaScript, and static assets

Generated: February 1, 2026
System Version: 1.0
Python: 3.14.2 | *Flask: 3.1.2*