# Detailed Documentation for Text-to-Video Prompt Processor

Team 8

Team Members:

Hema Srita

Harshdeep

Ishaan CP

Adit

## 1. Setup Instructions

### 1.1 Prerequisites

Ensure the following are set up before running the project:  
- Hardware: A system with GPU support is strongly recommended for faster processing (NVIDIA GPUs preferred).  
- Software  
 - Python version: 3.8 or higher  
 - Dependencies: `openai`, `diffusers`, `transformers`, `accelerate`, `gradio`  
- API Keys: Access to NVIDIA's APIs such as Neva, Kosmos, or Phi-3 is required.

### 1.2 Installation Steps

1. Install the required libraries:  
 ```bash  
 pip install openai  
 pip install git+https://github.com/huggingface/diffusers transformers accelerate  
 pip install gradio  
 ```  
2. Ensure that all dependencies install without errors.

## 2. How to Use

### 2.1 Running the Application

1. Open a terminal and navigate to the project folder.  
2. Run the script:  
 bash  
 python main.py  
   
3. A URL will be displayed in the terminal, similar to:  
   
 Running on local URL: http://127.0.0.1:7860/  
   
 Open this link in a browser to access the Gradio interface.

### 2.2 Using the Interface

1. Upload a Video:  
 - Use the "Upload" button in the Gradio interface to upload a video file.  
2. Process Video:  
 - Click the "Generate" button to analyze the video and process its frames.  
3. View Results:  
 - The system will generate:  
 - A Textual Description (Prompt) summarizing the content of the video.  
 - A Generated Video based on the detected features in the original video.

## 3. System Architecture

### 3.1 Components

1. Input Video Handling:  
 - Frames are extracted using OpenCV.  
 - Only every 30th frame is processed to optimize performance.  
2. Visual Language Model (VLM):  
 - NVIDIA Neva or similar VLMs are used to analyze individual frames.  
 - Key data extracted:  
 - Gender of the subject (e.g., male or female).  
 - Facial expressions (e.g., smiling, laughing, neutral).  
3. Prompt Aggregation:  
 - Textual outputs from the VLM for multiple frames are aggregated into a consolidated prompt.  
4. Text-to-Video Diffusion Model:  
 - The consolidated prompt is passed into a diffusion-based generative model (Hugging Face's `text-to-video-ms-1.7b).  
 - Outputs are saved as frames, which are then combined into a video.  
5. Interactive Interface:  
 - Gradio provides an intuitive interface for uploading videos and visualizing results.

### 3.2 Workflow

1. Input Video: Uploaded via Gradio.  
2. Frame Processing: Analyze frames using NVIDIA Neva VLM.  
3. Generate Prompt: Text summarizing frame analysis is created.  
4. Synthesize Video: Use the prompt to create a new video.  
5. Output: Results displayed in the interface.