1. Project Overview

1.1 What is the Project?

Rewind is a Video Memory System that processes 24-hour video recordings to create a searchable memory database. It enables natural language queries to find objects, events, and conversations from past recordings. The system combines computer vision, audio transcription, and multimodal AI to create a searchable archive of daily life.

Core Concept: Like a personal assistant that remembers everything you see and hear, allowing you to ask questions like "Where did I last see my water bottle?" or "When was my CS366 exam mentioned?"

1.2 Key Features

- Multimodal Processing: Processes both video frames and audio transcripts
- Object Detection: Identifies objects (cup, phone, person, laptop, keys) using YOLOv8
- Audio Transcription: Transcribes speech using Whisper with word-level timestamps
- Semantic Search: Uses CLIP embeddings for natural language queries
- Conversational AI: Generates natural language responses about memories
- Time Tracking: Maps video timestamps to actual calendar dates/times
- Dual Query Types: Supports both object queries (with images) and memory/event queries (text-only)

2. What Has Been Completed

2.1 Core Infrastructure (V1)

Video Processing Pipeline:
- Frame extraction at configurable intervals (default: 0.75s)
- YOLOv8 object detection on each frame
- CLIP embedding generation for visual search
- FAISS vector store for efficient similarity search
- Metadata storage with timestamps and object detections

Query System:
- Text-based semantic search using CLIP embeddings
- Image-based search capability
- Basic response generation

2.2 Enhanced System (V2) - Still needs to be tested

24-Hour Video Chunk Processing:
- Frame extraction at 5-10 fps (configurable, default: 7.5 fps)
- Handles full 24-hour video recordings from Raspberry Pi
- Date/time tracking with absolute datetime mapping
- Video metadata management with period tracking

Audio Transcription Integration:
- FFmpeg audio extraction from video files
- Whisper transcription with word-level timestamps
- Separate vector store for audio transcripts
- Time synchronization between video and audio

Advanced Query System:
- Dual-mode search: video frames and audio transcripts
- Query classification: automatically detects object vs. memory queries
- Conversational AI responses using open-source LLMs (HuggingFace)
- Frame extraction for object queries (saves actual frame images)
- Smart template fallback when LLM APIs are unavailable

Query Types Implemented
- Object Queries: "Where is my water bottle?" → Returns frame image + conversational response
- Memory Queries: "When was CS366 midterm mentioned?" → Returns text-only response with formatted dates

3. Current Work

3.1 What I'm Working on Right Now

System Refinement:
- Optimizing frame extraction rates for 24-hour videos (balancing accuracy vs. processing time)
- Improving query classification accuracy
- Enhancing conversational responses with better context understanding
- Testing with real-world video data

Performance Optimization:
- Reducing processing time for 24-hour videos
- Optimizing vector store search performance

- Improving memory usage for large video datasets

User Experience:
- Refining conversational responses to be more natural
- Improving error handling and edge cases
- Adding better logging and progress indicators

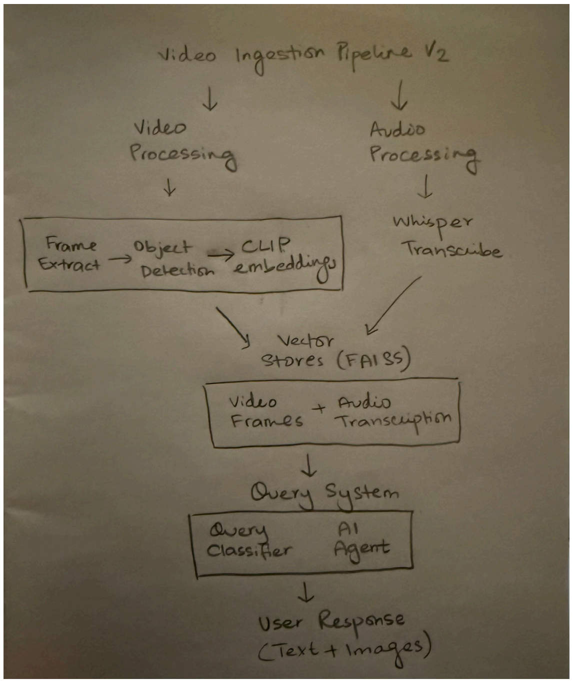
4. Plan Before Final Demo

1. Demo Preparation
   - Create sample video dataset with diverse scenarios
   - Prepare example queries showcasing both object and memory queries
   - Generate output examples (frame images, responses)
   - Create visualizations of system architecture

2. UI/Interface Improvements
   - Build simple command-line interface for demo

3. Query Response Quality
   - Fine-tune conversational responses
   - Improve date/time formatting
   - Add more context to responses

5. System Architecture



6. Other Details

6.1 Technologies Used

Computer Vision:

- YOLOv8 (Ultralytics): Real-time object detection
- CLIP (OpenAI): Multimodal embeddings for images and text

Audio Processing:

- Whisper (OpenAI): Speech-to-text transcription
- FFmpeg: Audio extraction from video

Vector Search:

- FAISS (Facebook AI Similarity Search): Efficient similarity search
- 512-dimensional embeddings (CLIP base model)

AI/LLM:

- HuggingFace Inference API: Open-source LLM access
- Template-based fallback: Smart responses without API

6.2 Audience Engagement Plan

1. Live Demo - Object Query
   - Query: "Where is my water bottle?"
   - Show: Real-time search → Frame extraction → Conversational response
   - Highlight: Natural language understanding

2. Live Demo - Memory Query
   - Query: "When was my CS366 exam mentioned?"
   - Show: Audio search → Date extraction → Formatted response
   - Highlight: Multimodal search capability

3. Technical Deep Dive
   - Show system architecture diagram
   - Explain: YOLOv8 detection, CLIP embeddings, Whisper transcription
   - Show: Vector store statistics, processing pipeline

5. Interactive Demo
   - Allow audience to suggest queries
   - Show real-time processing

7. References

1. Redmon, J., et al. (2016). "You Only Look Once: Unified, Real-Time Object Detection." CVPR 2016.

2. Radford, A., et al. (2021). "Learning Transferable Visual Models From Natural Language Supervision." ICML 2021.

3. Radford, A., et al. (2022). "Robust Speech Recognition via Large-Scale Weak Supervision." arXiv:2212.04356.

4. Johnson, J., et al. (2019). "Billion-scale similarity search with GPUs." arXiv:1702.08734.