Badminton AI Analysis: LangGraph Orchestration Pipeline

ShreeRaj Mummidivarapu

June 13, 2025

What is Agentic AI?

- **Traditional ML/AI Systems (e.g., CNNs, Vision Transformers, LLMs):**
  - **CNNs (Convolutional Neural Networks):** Excellent for image classification, object detection. Primarily pattern recognition.
  - **Vision Transformers (ViTs):** Leverage self-attention for image tasks, capturing global dependencies. Still largely reactive.
  - **LLMs (Large Language Models):** Powerful for text generation, understanding. Can exhibit emergent reasoning but lack inherent agency or persistent state.
  - **Common Characteristics:**
    - *Monolithic & Reactive:* Designed for specific tasks, respond to input without internal goals or long-term planning.
    - *Limited Self-Correction:* Require retraining for significant behavioral changes.
    - *No Persistent State:* Each interaction is often independent, lacking memory across sessions.

Agentic AI Systems

- **Composed of autonomous agents with specific roles.**
- Possess capabilities such as:
    - Planning
    - Memory
    - Tool use
    - Self-reflection
- Can break down complex tasks, orchestrate actions, and adapt to new information.
- Aim for more human-like problem-solving, decision-making, and continuous learning.
- **Key Capabilities:** Planning, memory, tool use, self-reflection, and dynamic adaptation.
- **Examples:** Autonomous research agents, complex task automation systems, adaptive control systems.

Why Agentic AI for Multimodal Sports Analysis?

- **Complexity of Sports Analysis:**
  - Requires understanding of visual (player movement, shuttlecock trajectory) and audio (shuttlecock hit, player grunts) cues.
  - Contextual understanding: game state, player strategy, real-time dynamics.
  - Traditional models struggle with integrating diverse data streams and dynamic reasoning.
- **Agentic AI Advantages:**
  - **Modularity:** Separate agents can specialize in video processing, audio analysis, and strategic interpretation.
  - **Orchestration:** LangGraph enables seamless flow of information and decision-making between agents.
  - **Adaptability:** Agents can learn and refine their understanding based on ongoing analysis and feedback.
  - **Holistic View:** Combines disparate data points into a coherent, actionable understanding of the match.

Problem: The Coaching Gap

- **Lack of Objective, Granular Feedback:**
  - Human coaches, while invaluable, can miss subtle technical flaws or strategic patterns due to the speed and complexity of badminton.
  - Feedback is often subjective and not consistently data-driven.
- **Limited Accessibility to Elite Analysis:**
  - High-quality, personalized coaching and performance analysis are often expensive and inaccessible to amateur players or those in underserved regions.
- **Inefficient Performance Tracking:**
  - Manual analysis is time-consuming and prone to human error, making long-term performance tracking and progress assessment challenging.

Solution: LangGraph Pipeline for Badminton Analysis (1/2)

- **Multimodal Data Integration:**
  - Effective integration of video analysis (pose metrics) and audio transcription in a unified pipeline.
  - Sequential processing with comprehensive error handling for reliable results.
- **LangGraph for Sports Analysis:**
  - Pioneering the use of LangGraph for structured pipeline orchestration in sports analytics.
  - Foundation for future expansion to more complex agent interactions and workflows.
- **Actionable, Granular Feedback:**
  - Focus on generating highly specific, actionable feedback for players and coaches.
  - Moves beyond descriptive statistics to prescriptive recommendations.
- **Scalable & Extensible Architecture:**
  - Modular pipeline design allows for easy addition of new analysis capabilities in future iterations.

Solution: LangGraph Pipeline for Badminton Analysis (2/2)

- **Holistic System View:**
    - Our system seamlessly combines video (player movement) and audio (speech transcription) for comprehensive understanding.
- **Linear Pipeline Orchestration:**
    - Utilizes a four-node LangGraph pipeline for video processing, audio processing, and report generation.
    - Sequential processing with state management for efficient data flow between components.
- **Automated, Granular Reporting:**
    - Generates detailed, objective reports with actionable insights.
    - Identifies strengths and areas for improvement to support targeted coaching decisions.

How It Works: High-Level Overview

Input:

- Badminton match video.

Processing:

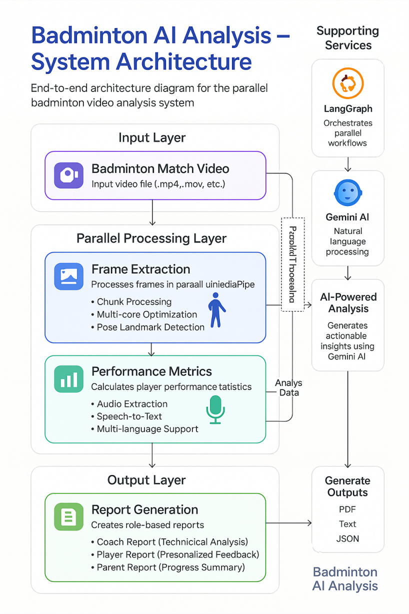
- Video frames are processed for pose estimation and audio is transcribed for speech content.
- LangGraph orchestrates a linear pipeline with four processing nodes for efficient data flow.

Output:

- Comprehensive text and PDF reports with actionable insights and strategic recommendations.

System Pipeline Overview

System Architecture Diagram



Badminton AI Analysis – System Architecture

End-to-end architecture diagram for the parallel badminton video analysis system

Input Layer

Badminton Match Video
Input video file (.mp4,.mov, etc.)

Parallel Processing Layer

Frame Extraction
Processes frames in paraall uniediaPipe

- Chunk Processing
- Multi-core Optimization
- Pose Landmark Detection

Performance Metrics
Calculates player performance tatistics

- Audio Extraction
- Speech-to-Text
- Multi-language Support

Output Layer

Report Generation
Creates role-based reports

- Coach Report (Technicical Analysis)
- Player Report (Presonalized Feedback)
- Parent Report (Progress Summary)

Supporting Services

LangGraph
Orchestrates parallel workflows

Gemini AI
Natural language processing

AI-Powered Analysis
Generates actionable insights using Gemini AI

Generate Outputs
PDF
Text
JSON

System Pipeline: High-Level Walkthrough (1/2)

- **1. Video Input:**
  - Raw match footage (e.g., MP4 files).
  - Supported formats: MP4, AVI, MOV with H.264/H.265 encoding.
  - Optimal resolution: 1080p (1920×1080) at 60fps for detailed motion capture.
- **2. Frame & Audio Extraction:**
  - Video is processed to extract individual frames for visual analysis.
  - Audio track is separated and processed for speech transcription.
  - Preprocessing includes frame resizing and color conversion.

System Pipeline: High-Level Walkthrough (2/2)

- **3. Pipeline Orchestration with LangGraph:**
  - Extracted data flows through a four-node linear pipeline orchestrated by LangGraph.
  - Pipeline includes video processing, audio transcription, data integration, and report generation.
  - State management ensures efficient data flow between processing steps.
- **4. Report Generation:**
  - Consolidated data is processed by Google Gemini API to generate comprehensive text and PDF reports.
  - Reports include performance metrics, observations based on pose data, and actionable feedback.
  - Customizable templates based on user role (player, coach, analyst).

Detailed Technical Implementation

Video Processing: Technical Details (1/2)

- **Frame Extraction Pipeline:**
  - **Decoding:** OpenCV (`cv2.VideoCapture`) decodes video stream into raw frame data.
  - **Sampling:** Configurable frame sampling (default: every 5th frame) for efficient processing.
  - **Preprocessing:** Resize to 640×360, convert from BGR to RGB format for MediaPipe compatibility.
- **Pose Estimation Implementation:**
  - **Model:** MediaPipe Pose with `static_image_mode=True` for independent frame processing.
  - **Configuration:** `model_complexity=1`, `detection_confidence=0.5`, `tracking_confidence=0.5`.
  - **Keypoints:** Extracts 7 key landmarks including nose, wrists, elbows, and shoulders.

Video Processing: Technical Details (2/2)

- **Performance Metrics Calculation:**
  - **Wrist Distance:** Euclidean distance between left and right wrist keypoints.
  - **Elbow Angles:** Angle calculation between shoulder, elbow, and wrist points for both arms.
  - **Data Structure:** Results stored as structured JSON with frame number, timestamp, keypoints, and metrics.

Audio Processing: Extraction and Transcription

- **Audio Extraction and Preprocessing:**
  - **Extraction:** `PyDub` library separates the audio track from the video file and saves it as WAV format.
  - **Optimization:** Audio is converted to mono and 16kHz to reduce size and improve processing speed.
  - **Segmentation:** Audio stream is divided into chunks based on silence detection (500ms threshold).
- **Speech Recognition:**
  - **Processing:** Each audio chunk is processed individually to improve transcription accuracy.
  - **Transcription:** Google's Web Speech API is used for converting speech to text.
  - **Language Support:** Multiple language options are available for international players and coaches.

Audio Processing: Current & Future Capabilities

- **Current Implementation Note:**
  - Focuses on speech transcription, not sound event detection.
  - Basic pipeline supports chunk-wise transcription for improved accuracy.
- **Future Enhancements:**
  - Integrate shuttlecock hit detection via amplitude and spectral spike analysis.
  - Add player vocalization detection to analyze intensity and engagement.
  - Explore use of Whisper or custom-trained audio classification models.

LangGraph Pipeline: Architecture and Integration

- **Pipeline Architecture:**
  - Linear processing pipeline with four main nodes:
    - Video processing
    - Audio processing
    - Data integration
    - Report generation
  - Simple Directed Acyclic Graph (DAG) with sequential flow and error-handling edges.
  - Supports both synchronous and asynchronous execution using `asyncio`.
- **Integration with Python Ecosystem:**
  - Integrates with `asyncio` for non-blocking execution.
  - Compatible with:
    - MediaPipe (vision)
    - Google Web Speech API (audio)
    - Gemini API (report generation)

LangGraph Pipeline: Technical Implementation

- **Graph Definition:**
    - Built using a simple `StateGraph` with four processing nodes and defined edges.
- **State Management:**
    - Uses a custom `TypedDict` called `BadmintonState` to carry data between stages.
- **Execution Model:**
    - Sequential execution with robust error handling at each node.
- **Resource Optimization:**
    - Dynamically adjusts worker allocation based on available CPU cores and memory.
- **Progress Tracking:**
    - Built-in progress reporting and logging for long-running operations.

Report Generation: Technical Implementation

- **Data Aggregation:**
  - **Input Sources:** Combines pose metrics from video analysis and speech transcription from audio.
  - **Data Sampling:** Processes first 100 pose metrics to manage context size for LLM processing.
  - **JSON Formatting:** Structures data in standardized format for AI model consumption.
- **Natural Language Generation:**
  - **AI Model:** Google's Gemini 1.5 Flash model generates contextual, role-specific reports.
  - **Role-Based Prompting:** Custom system prompts tailored to coach, student, or parent perspectives.
  - **Personalization:** Adapts language, technical depth, and focus areas based on target audience.
- **Multilingual Support:**
  - **Language Options:** Reports available in multiple languages (English, Hindi, Tamil, Telugu, Kannada).

Key Components: Vision and Audio Nodes

- **Video Processing Node:**
  - **Primary Function:** Analyzes video frames for player pose
  - **Technical Implementation:** MediaPipe Pose model for human pose detection and tracking.
  - **Key Capabilities:** Pose estimation, elbow angle calculation, wrist distance measurement.
  - **Output:** Structured JSON with timestamped keypoints and performance metrics.
- **Audio Processing Node:**
  - **Primary Function:** Extracts and transcribes speech from the video's audio track.
  - **Technical Implementation:** PyDub for audio extraction; Google Web Speech API for transcription.
  - **Key Capabilities:** Multi-language support, silence-based segmentation.
  - **Output:** Transcribed text of spoken content.

Key System Components: Pipeline Nodes

- **Report Generator:**
  - **Primary Function:** Synthesizes pose metrics and transcription data into coherent, actionable feedback.
  - **Technical Implementation:** Google's Gemini 1.5 Flash model with role-specific prompting strategies.
  - **Key Capabilities:** Multilingual report generation, role-based content adaptation, contextual analysis.
  - **Adaptation Logic:** Tailors content depth, technical terminology, and focus areas based on user role (coach, student, parent).
  - **Output:** Text-based report with structured sections appropriate to the target audience.

Why LangGraph? Architectural Advantages

- **State-aware Execution:**
  - Unlike Airflow (designed for static DAGs), LangGraph supports stateful execution graphs.
  - Maintains state across processing steps, allowing for efficient data flow between components.
  - Enables structured error handling and recovery mechanisms.
- **Pipeline Orchestration Benefits:**
  - Built for orchestrating complex processing workflows with clear separation of concerns.
  - Provides efficient state management through TypedDict implementation.
  - Supports future expansion to more complex workflows and agent-based systems.

Why LangGraph? Technical Implementation

- **Flexibility & Iteration:**
  - Enables structured pipeline development with clear node separation and error handling.
  - Supports both synchronous and asynchronous execution models through asyncio integration.
- **Integration with Python Ecosystem:**
  - Seamlessly integrates with asyncio for non-blocking operations.
  - Compatible with MediaPipe for vision processing and Google Web Speech API for audio transcription.
  - Works efficiently with Gemini API for report generation.
- **Technical Implementation Details:**
  - **Graph Definition:** Simple StateGraph with four nodes and edges.
  - **State Management:** BadmintonState TypedDict maintains data across processing steps.
  - **Execution Model:** Sequential processing with comprehensive error handling.
  - **Resource Optimization:** Dynamic worker allocation based on available CPU cores and memory.

Novelty and Innovation

Multimodal Data Integration:

- Effective integration of video analysis (pose metrics) and audio transcription in a unified pipeline.
- Sequential processing with comprehensive error handling for reliable results.

LangGraph for Sports Analysis:

- Pioneering the use of LangGraph for structured pipeline orchestration in sports analytics.
- Foundation for future expansion to more complex agent interactions and workflows.

Actionable, Granular Feedback:

- Focus on generating highly specific, actionable feedback for players and coaches.
- Moves beyond descriptive statistics to prescriptive recommendations.

Scalable & Extensible Architecture:

- Modular pipeline design allows for easy addition of new analysis capabilities in future iterations.

System Implementation: Technical Stack

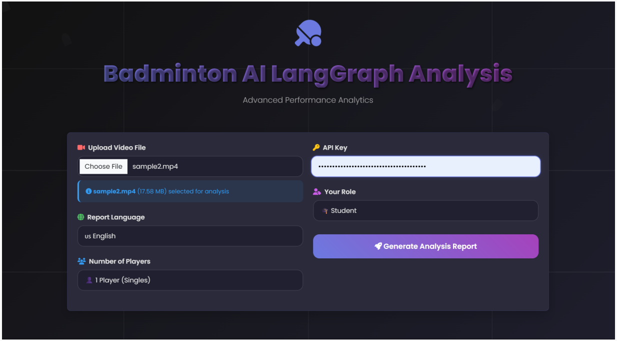
Frontend Technologies:

- **Web Interface:** HTML5, CSS3, JavaScript with responsive design.
- **Video Upload:** Custom file uploader with format validation and progress tracking.
- **Report Viewer:** Interactive PDF viewer with annotation capabilities.

Backend Technologies:

- **Server:** Flask for web application serving and file handling.
- **Video Processing:** OpenCV, MediaPipe for pose estimation and tracking.
- **Audio Processing:** Google Web Speech API for audio transcription.
- **Pipeline Orchestration:** LangGraph for linear pipeline definition and state management.
- **LLM Integration:** Google Gemini API for report generation with custom prompt templates.
- **Report Generation:** Custom templates for role-based and multilingual reports.

Demo Step 1: Upload Video



Badminton AI LangGraph Analysis

Advanced Performance Analytics

Upload Video File

Choose File sample2.mp4

sample2.mp4 (17.58 MB) selected for analysis

Report Language

us English

Number of Players

1 Player (Singles)

API Key

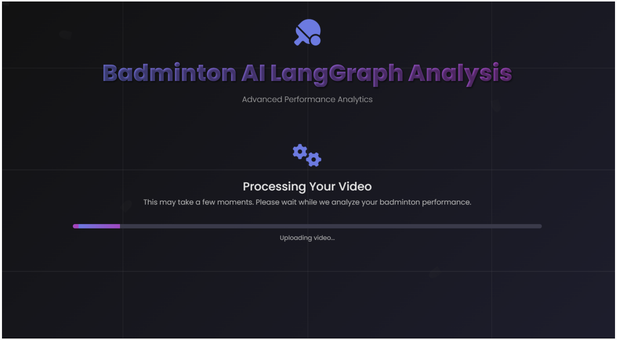
••

Your Role

Student

- User uploads a badminton match video and all necessary through the web interface.
- Supported formats include MP4, AVI, MOV.
- The system initiates pre-processing after recieving all information

Demo Step 2: Processing Status



Badminton AI LangGraph Analysis

Advanced Performance Analytics

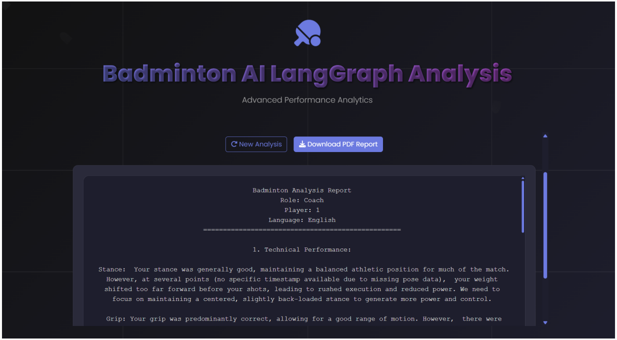
Processing Your Video

This may take a few moments. Please wait while we analyze your badminton performance.

Uploading video...

- System displays processing stages with real-time progress indicators.
- Separate modules handle video frame extraction, pose detection, and audio transcription.
- Users are informed of each module's completion status.

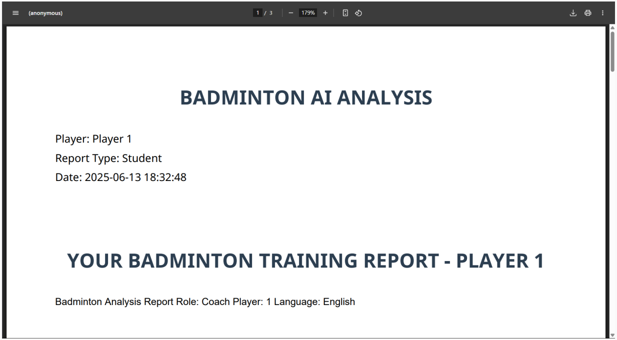
Demo Step 3: Interactive Report


Badminton AI LangGraph Analysis
Advanced Performance Analytics
New Analysis
Download PDF Report

```
Badminton Analysis Report
Role: Coach
Player: 1
Language: English
==================================================

1. Technical Performance:

Stance:  Your stance was generally good, maintaining a balanced athletic position for much of the match.
However, at several points (no specific timestamp available due to missing pose data),  your weight
shifted too far forward before your shots, leading to rushed execution and reduced power. We need to
focus on maintaining a centered, slightly back-loaded stance to generate more power and control.

Grip: Your grip was predominantly correct, allowing for a good range of motion. However,  there were
```

- An interactive, browser-based report is generated after processing.
- Key metrics like elbow angles, wrist distances, and speech insights are displayed.
- Visual overlays and summaries make interpretation intuitive.

Demo Step 4: Download PDF Report



BADMINTON AI ANALYSIS

Player: Player 1
Report Type: Student
Date: 2025-06-13 18:32:48

YOUR BADMINTON TRAINING REPORT - PLAYER 1

Badminton Analysis Report Role: Coach Player: 1 Language: English

- Users can explore specific sections in detail through the interface.
- A downloadable PDF report summarizes key findings for offline review.
- Report layout adapts to role — coach, player, or parent.

Demo Walkthrough: User Experience Summary

- **Complete Journey Overview:**
  - **Step 1: Upload** — Upload badminton match video via a simple interface.
  - **Step 2: Process** — Monitor real-time progress with clear indicators.
  - **Step 3: Report View** — See insights in a rich, interactive web view.
  - **Step 4: Export** — Download professional reports for long-term use.
- **Focus on Usability:** Designed for non-technical users (coaches, athletes).
- **Future Enhancement:** Mobile-friendly UI and real-time streaming support.

Thank You! Questions?