Comprehensive Technical Feasibility and Architectural Specification: Next-Generation Multimodal Autonomous Agents

1. Executive Strategy and Architectural Convergence

The contemporary landscape of artificial intelligence is undergoing a seismic shift from unimodal, task-specific models toward unified, multimodal foundation models that approximate general-purpose perception. The project currently under development aims to synthesize this convergence into a coherent agentic system capable of reasoning across visual, auditory, and textual domains with high temporal fidelity. Based on a rigorous analysis of the latest research literature—spanning late 2024 through early 2025—the proposed architecture sits at the intersection of four critical technological breakthroughs: **Promptable Concept Segmentation** (exemplified by Meta's SAM 3), **Sigmoid-Loss Semantic Encoding** (Google's SigLIP 2), **Hierarchical Temporal Compression** (OpenGVLab's InternVideo2.5), and **Unified Audio-Visual Reasoning** (Alibaba's Qwen2.5-VL and Qwen2-Audio).

This report provides an exhaustive technical analysis of these components, evaluating their interoperability, architectural innovations, and deployment constraints. The central thesis driving this specification is that the traditional "memory-resolution trade-off"—where models must sacrifice fine-grained spatial detail to handle long temporal contexts—has been effectively neutralized by the simultaneous emergence of **Naive Dynamic Resolution** in Vision-Language Models (VLMs) and **Hierarchical Token Compression (HiCo)** in video foundation models. By integrating these specific technologies, the proposed project can achieve a level of "open-world" understanding that was computationally infeasible just months prior.

The following sections dissect the specific capabilities of each subsystem. We will explore how SAM 3's unified detector-tracker architecture enables open-vocabulary concept discovery without retraining. We will analyze the necessity of SigLIP 2's sigmoid-based training objective for high-fidelity dense retrieval. We will examine the critical role of long-context video modeling via InternVideo2.5 to handle temporal dependencies spanning minutes or hours. Finally, we will address the auditory dimension via Qwen2-Audio and the necessity of specialized Optical Character Recognition (OCR) subsystems like PaddleOCR for dense text extraction, culminating in a unified instruction-tuning strategy that binds these modalities into a single, responsive agent.

2. The Visual Perception Subsystem: Segment Anything Model 3 (SAM 3)

The visual core of the proposed system must move beyond the limitations of "Promptable Visual Segmentation" (PVS)—where users manually identify targets via clicks or boxes—toward "Promptable Concept Segmentation" (PCS). The release of **Segment Anything Model 3 (SAM 3)** in November 2025 marks a definitive transition in this domain, introducing an architecture capable of autonomously resolving semantic concepts into pixel-perfect masks.[1]

2.1. Paradigm Shift: From Visual Prompts to Concept Prompts

The defining innovation of SAM 3 is its ability to accept open-vocabulary text prompts (e.g., "red helmet") or exemplar images and autonomously detect, segment, and track all corresponding instances within a visual feed.[1] This contrasts sharply with its predecessors. In SAM 2, a user seeking to segment multiple elephants in a video would need to provide explicit visual prompts (clicks) for each distinct elephant. SAM 3 eliminates this bottleneck by internalizing the concept of "elephant," allowing the model to perform exhaustive instance segmentation based on a single semantic query.[2]

This capability relies on a unified model architecture that integrates detection, segmentation, and tracking. Unlike previous pipelined approaches that might chain a YOLO detector to a SAM segmentation module—incurring double the computational cost for encoding—SAM 3 utilizes a shared **Perception Encoder (PE)**.[3] This vision backbone processes the incoming image or video frame exactly once, generating a set of feature embeddings referred to as "unconditioned tokens".[3] These tokens serve as the single source of truth for all downstream tasks, dramatically optimizing inference throughput.

2.2. Architectural Granularity: The Perception Encoder and Presence Token

The Perception Encoder (PE) is designed with windowed attention mechanisms that balance computational efficiency with the need for global context. It divides the image into non-overlapping windows to manage the quadratic complexity of standard transformers while applying global attention layers at specific intervals to capture long-range dependencies.[3] This dual-nature attention ensures that the model can segment fine details (like the rim of a wheel) while understanding the global structure of the object (the car itself).

A critical innovation within the detection head is the **Presence Token**. In open-vocabulary segmentation, a major source of error is "hallucination," where the model forces a segmentation mask even when the target concept is absent. SAM 3 introduces a learned global token that predicts a binary probability of the concept's presence *before* the model attempts localization.[1] This decouples the "recognition" task (Is it there?) from the "localization" task (Where is it?), resulting in significantly higher precision in negative scenarios. This presence head essentially gates the segmentation process, ensuring computational resources are not wasted generating masks for non-existent objects.[1]

2.3. Quantitative Performance and the "Human Gap"

The performance of SAM 3 is quantified primarily through the **SA-Co (Segment Anything with Concepts)** benchmark, a rigorous evaluation suite designed to measure open-vocabulary competence. The metrics indicate that SAM 3 is not merely an incremental update but a functional leap in bridging the gap between automated segmentation and human-level perception.

Table 1: SAM 3 Performance Metrics Comparison [1]

Metric	**Description**	**SAM 3 Achievement**	**Previous Best (SOTA)**	**Improvement**
LVIS Zero-Shot Mask AP	Zero-shot segmentation accuracy on rare classes.	**47.0**	38.5	+22.1%
SA-Co Benchmark (CGF1)	Classification-Gated F1 score for concept segmentation.	**65.0**	34.3 (OWLv2)	+89.5%
ADE-847 (mIoU)	Mean Intersection over Union on semantic segmentation.	**14.7**	9.2 (APE-D)	+59.8%
Cityscapes (mIoU)	Semantic segmentation accuracy on urban scenes.	**65.1**	44.2 (APE-D)	+47.3%
Inference	Latency per	**30 ms**	N/A	Real-time

Speed	image on H200 GPU.			capable

The **CGF1 (Classification-Gated F1)** metric is particularly revealing. It combines positive macro F1 (pmF1), which measures localization quality, with Image-Level Matthews Correlation Coefficient (IL_MCC), which measures binary classification accuracy. The 89.5% improvement over OWLv2 underscores the effectiveness of the Presence Token and the unified PE architecture.[1] Furthermore, reports indicate that SAM 3 achieves 88% of the estimated human lower bound on the SA-Co/Gold dataset, suggesting that the model's error rate is approaching the level of inter-annotator disagreement found among human labelers.[1]

2.4. Video Tracking and Memory Mechanisms

For the video domain, SAM 3 inherits and refines the memory mechanisms of SAM 2. The system maintains a memory bank that stores feature representations of objects across frames. This allows the model to handle occlusions—where an object temporarily disappears behind an obstacle—by referencing its "memory" of the object's appearance from previous frames.[2]

Crucially, SAM 3 implements a **"periodic re-prompting"** strategy.[3] In long-duration videos, tracker drift is inevitable as the object's appearance changes due to lighting, angle, or deformation. SAM 3 uses the high-confidence outputs of its image-level detector to periodically "reset" the tracker. This creates a symbiotic relationship: the tracker provides temporal continuity, while the detector provides semantic grounding, preventing the system from drifting onto a similar-looking but incorrect object (e.g., switching from one zebra to another in a herd). This integration is seamless because both components draw from the same Perception Encoder tokens, requiring no redundant feature extraction.[3]

3. The Semantic Embedding Subsystem: SigLIP 2

While SAM 3 excels at isolating pixels, the project requires a robust semantic engine to understand what those pixels represent in a high-dimensional vector space. **SigLIP 2**, released by Google DeepMind in 2025, represents the current state-of-the-art for this "Semantic Encoder" role. It fundamentally rethinks the loss functions and training objectives of vision-language models to support better dense prediction and retrieval.[4]

3.1. Theoretical Advantage: Sigmoid Loss over Softmax

The original CLIP (Contrastive Language-Image Pre-training) architecture relied on a softmax contrastive loss, which forces the model to select the single "best" text description for a given image from a batch. This creates a zero-sum game: for the probability of "dog" to go up, the probability of "cat" must go down.

SigLIP 2 rejects this in favor of **Sigmoid Loss**. This approach treats every image-text pair as an independent binary classification problem.[5]

- **Non-Exclusive Multi-Labeling:** By using sigmoid loss, the model learns that a single image can validly match multiple distinct concepts (e.g., "sunny day," "beach," "family vacation") without those concepts competing against each other.
- **Dense Feature Quality:** This independence is crucial for "region-level" understanding. In a complex scene, a specific region might represent both "red car" and "traffic violation." Softmax would force a choice; Sigmoid allows both representations to coexist in the embedding space, leading to richer, more nuanced semantic vectors.[5]

3.2. Geometric Fidelity: The NaFlex Variant

A pervasive limitation in standard Vision Transformers (ViT) is the rigid requirement for square inputs (typically 224x224 or 256x256). This necessitates resizing or cropping, which introduces distortion (aspect ratio changes) or information loss (cropping out edges). For a project involving OCR or precise object detection, this distortion can be catastrophic—a "tall" receipt squeezed into a square becomes unreadable.

SigLIP 2 introduces the **NaFlex (Native Aspect Ratio and Flexible Resolution)** variant.[6]

- **Mechanism:** NaFlex allows the model to process images in their native aspect ratio by treating the image as a variable-length sequence of patches. It removes the need for padding or warping.
- **Performance Impact:** Evaluations on the Crossmodal-3600 retrieval benchmark show that NaFlex variants significantly outperform standard square-input models, particularly on tasks involving document understanding or panoramic scenes where geometry contains semantic meaning.[5] This ensures that the spatial relationships between objects—vital for agentic reasoning—are preserved in the embedding space.

3.3. Training Objectives for Dense Prediction

SigLIP 2 is not merely trained for global image classification. Its training recipe includes **decoder-based pretraining**, where the model must predict captions from image features.[5] This forces the encoder to retain fine-grained spatial information that might otherwise be discarded by a purely discriminative loss.

Furthermore, it employs **Self-Distillation and Masked Prediction**. Inspired by Masked Autoencoders (MAE), the student network is tasked with reconstructing the features of masked image patches based on the output of a teacher network.[6] This objective explicitly trains the model to understand local context and texture, which is indispensable for dense prediction tasks like depth estimation and semantic segmentation. The combination of these objectives allows SigLIP 2 to serve as a powerful feature extractor for the Region Encoder

Network (REN), bridging the gap between SAM 3's masks and the MLLM's text reasoning.[8]

3.4. Multilingual and Bias Mitigation

For a globally deployable project, language coverage is essential. SigLIP 2 is trained on the WebLI dataset, comprising 10 billion images and 12 billion alt-texts across 109 languages.[9] Crucially, the training data is curated to include de-biasing techniques, which notably reduce unfair object-to-gender associations (e.g., dissociating "doctor" from male figures).[5] This "Active Data Curation" (ACID) ensures that the semantic embeddings are not only accurate but also ethically robust, a requirement for any modern production system.

---

4. Temporal Reasoning and Long-Form Video: InternVideo2.5

Video is fundamentally distinct from static imagery due to the dimension of time. A simple "bag of frames" approach fails to capture causality, motion, and long-term dependencies. **InternVideo2.5**, leveraging the **Hierarchical Token Compression (HiCo)** mechanism, addresses the specific challenge of modeling long videos (minutes to hours) without overwhelming the context window of the reasoning LLM.[10]

4.1. The Context Bottleneck in Video-LLMs

Standard Video-LLMs often sample a few frames (e.g., 8 or 16) from a video to represent its content. While computationally cheap, this approach is "temporally blind." It cannot answer questions like "Did the car turn left *before* or *after* the light turned red?" because the intermediate frames containing the turn might be skipped. Conversely, inputting every frame generates an explosion of tokens—a 2-minute video at 1 FPS generates thousands of tokens, saturating the context window and slowing inference to a crawl.[12]

4.2. The HiCo Solution: Hierarchical Compression

InternVideo2.5 solves this via **Hierarchical Token Compression (HiCo)**, a two-stage process designed to balance fidelity with efficiency.[11]

1. **Stage 1: Clip-Level Compression:** The long video is segmented into short clips. For each clip, a vision encoder extracts dense visual tokens. HiCo then compresses these tokens based on *spatiotemporal redundancies*. If a clip shows a static background with minor movement, HiCo retains only the changing features, discarding the redundant static tokens.
2. **Stage 2: Video-Level Compression:** The compressed tokens from all clips are aggregated. A second compression pass is applied, often guided by the user's text query or semantic correlations. This creates a compact "video-level" representation that fits

within the LLM's context window while retaining the essential temporal markers.[11]

Impact: This architecture allows InternVideo2.5 to process context windows that are effectively **6x longer** than standard models.[14] It enables the model to "memorize" and reason about events occurring over extended durations, achieving state-of-the-art results on benchmarks like **VideoMME** (64.9% accuracy) and **LongVideoBench** (59.6% accuracy).[15]

4.3. Progressive Training Pipeline

The robustness of InternVideo2.5 stems from a three-stage progressive training pipeline [16]:

- **Stage 1 (Unmasked Reconstruction):** The model learns basic spatiotemporal structures by reconstructing masked video tubes (3D patches), similar to VideoMAE. This builds a strong "physics" understanding of motion.
- **Stage 2 (Cross-Modal Alignment):** The model is aligned with text using massive video-text datasets (e.g., InternVid2, WebVid). This connects visual motion to semantic concepts (e.g., connecting the motion of "waving" to the word "waving").
- **Stage 3 (Next-Token Prediction):** The model is fine-tuned as a dialogue agent, predicting the next text token given a video context. This enables the "chat" capability essential for an interactive agent.[17]

The dataset for this training is colossal, comprising **402 million data entries**, including 50 million video-audio-speech-text pairs from InternVid2.[16] This scale ensures the model generalizes well to "in-the-wild" footage, handling diverse lighting, camera angles, and action types.

5. Unified Multimodal Reasoning: Qwen2.5-VL and Qwen2-Audio

While InternVideo2.5 excels at long-form temporal compression, **Qwen2.5-VL** and **Qwen2-Audio** provide the "agentic" reasoning capabilities, particularly for tasks involving dynamic resolution, high-fidelity OCR, and complex auditory analysis.

5.1. Qwen2.5-VL: Naive Dynamic Resolution and Agentic Vision

Qwen2.5-VL introduces a mechanism termed **Naive Dynamic Resolution**.[18] Unlike models that force images into a fixed grid, Qwen2.5-VL dynamically allocates tokens based on the information density of the input.

- **Mechanism:** An image is processed in its native resolution. A complex, detail-rich region (like a document) generates many tokens, while a simple background region generates few. This is managed via **M-RoPE (Multimodal Rotary Position Embeddings)**, which effectively maps 2D spatial positions into the 1D token sequence required by the LLM.[18]

- **Implication:** This allows Qwen2.5-VL to act as a highly effective **Visual Agent**. It can operate devices (mobiles, robots) by reading screen text or identifying small UI elements that would be lost in a fixed-resolution downsampling. It supports videos over 20 minutes in length, making it a viable alternative or complement to InternVideo2.5 depending on the specific trade-off between resolution (Qwen's strength) and temporal compression (InternVideo's strength).[18]

5.2. Qwen2-Audio: The Auditory Cortex

A true multimodal agent must "hear" as well as see. **Qwen2-Audio** provides this capability by integrating the **Whisper-large-v3** encoder directly into the Qwen language model.[19]

- **Dual-Mode Operation:** Qwen2-Audio operates in two modes without requiring special system prompts:
  1. **Voice Chat Mode:** The user speaks, and the model responds. This is standard ASR + LLM.
  2. **Audio Analysis Mode:** The model analyzes the audio *signal* itself. It can identify non-speech events (e.g., "glass breaking," "siren wailing," "typing on a keyboard").[20]
- **Technical Pipeline:** Audio is resampled to 16kHz and converted to 128-channel log-mel spectrograms. A pooling layer compresses the sequence so that each encoder frame represents approximately 40ms of audio.[20] This encoder output is projected into the LLM's embedding space, allowing the model to "read" audio tokens alongside text tokens.
- **Why Whisper-large-v3?** The choice of Whisper-large-v3 ensures robust multilingual speech recognition and noise robustness, providing a solid foundation for the subsequent analysis layers.[19]

5.3. Safety and Robustness: SALMONN-Guard

For applications where safety is paramount, integrating a specialized safeguard model like **SALMONN-Guard** is recommended. Research indicates that standard Multimodal LLMs are vulnerable to "speech-audio composition attacks," where harmful instructions are hidden in background noise or complex audio mixtures.[21] SALMONN-Guard inspects speech, audio, and text jointly, reducing attack success rates to ~20%.[21] This component acts as a "pre-flight check" for audio inputs before they are processed by the core reasoning agent.

---

6. Text Extraction Subsystem: OCR Technologies

While Qwen2.5-VL possesses strong internal OCR capabilities, dedicated OCR pipelines are often required for specific engineering constraints, such as ultra-low latency or processing extremely dense technical documents embedded in video feeds (e.g., HUDs, scrolling logs).

6.1. Comparative Analysis: PaddleOCR vs. TrOCR

The choice of OCR engine dictates the system's latency profile.

Table 2: OCR Technology Comparison

Feature	**PaddleOCR (PP-OCRv4)**	**TrOCR (Transformer OCR)**	**EasyOCR**
Architecture	Two-stage: DBNet (Detection) + SVTR (Recognition).[22]	End-to-End Transformer (Encoder-Decoder) .[23]	CNN + LSTM + CTC.[24]
Speed	**Very High** (optimized for edge/CPU).[25]	**Low** (Computationally heavy due to transformer layers).[25]	Moderate (GPU accelerated).
Accuracy	High on standard text; robust to rotation.[22]	Excellent on handwritten/messy text; context-aware.[26]	Good for multilingual; weaker on complex layouts.
Use Case	Real-time video text (HUDs, subtitles).[27]	High-value document digitization; offline processing.	Quick prototyping.

- **PaddleOCR:** This is the recommended default for video-based text extraction. Its modular design allows it to run efficiently even on CPUs, processing images in milliseconds. The use of **DBNet** for detection ensures it handles curved or oriented text (common in natural scenes) effectively.[22]
- **TrOCR:** While more accurate for "messy" text, its latency makes it unsuitable for real-time video analysis. It is best reserved as a fallback system for high-resolution keyframes where PaddleOCR fails.[23]

7. System Integration and Instruction Tuning

The "brain" of the agent is formed by integrating these perception modules into a unified instruction-following system. This requires a specific data engineering pipeline known as **Multimodal Instruction Tuning**.

7.1. Data Engineering: The JSONL Structure

To train the MLLM to behave as a coherent agent, data must be structured in a specific JSON/JSONL format that interleaves modalities and explicitly defines "turns" in a conversation. Research from Video-LLaVA and InternVideo2 highlights the necessity of timestamp alignment.[28]

Recommended Data Structure:

JSON

```
{
  "id": "train_001",
  "video": "data/videos/vid_001.mp4",
  "conversations": <box_2d></box_2d>. The object is identified as a gift box due to the ribbon pattern."
    }
  ],
  "timestamps": [[10.0, 15.0]],
  "width": 1920,
  "height": 1080
}
```

- **Timestamp Grounding:** Explicitly annotating start/end times allows the model to learn temporal localization.
- **Bounding Box Tokens:** Including box coordinates (e.g., <box_2d>) in the text response trains the model to ground its textual answers in specific image regions, bridging the gap between "what" and "where".[30]

7.2. Region-Language Alignment: The REN Approach

To bridge the gap between SAM 3's pixel masks and the MLLM's text embeddings, the **Region Encoder Network (REN)** methodology is highly relevant.[8]

- **The Problem:** SAM 3 gives a mask. The MLLM gives text. How do we ensure the MLLM is talking about *that specific mask*?

- **The REN Solution:** REN uses a lightweight projector to convert the visual features from the masked region (extracted by SigLIP 2) into "region tokens." These tokens are injected into the MLLM's prompt. This allows the user to point to a region (via SAM 3) and ask the MLLM "What is this?" with high semantic fidelity.[8]

7.3. Training Strategies: Unsloth and Quantization

Training or fine-tuning these massive models requires efficient techniques.

- **Unsloth:** This framework optimizes the backpropagation of Llama/Qwen architectures, allowing for **2x faster training** and **60% less memory usage**.[31] It is essential for fine-tuning Qwen2.5-7B on consumer-grade hardware.
- **LoRA (Low-Rank Adaptation):** Instead of retraining all 7 billion parameters, LoRA freezes the main weights and trains small adapter layers. This reduces VRAM requirements for training from ~80GB to ~16GB, making it feasible on a single RTX 4090.[32]

8. Hardware Specifications and VRAM Budget

Deploying this multi-model architecture requires a precise understanding of memory consumption. The "15GB VRAM" often cited for a 7B model is a simplification.

8.1. Component VRAM Analysis

1. **SAM 3:** The model itself is likely large (estimated ~400MB-1GB for the weights), but the *activations* for high-resolution image processing can consume 4-6GB VRAM depending on the image size and number of masks.[1]
2. **Qwen2.5-7B (The Reasoning Core):**
   - **FP16 (Half Precision):** Requires ~16GB VRAM for weights alone. With KV cache (context memory), this easily exceeds 24GB.[33]
   - **4-bit Quantization:** Reduces weight memory to **~5.5GB**. This is the critical enabler for consumer deployment. With a reasonable context window (e.g., 8k tokens), the total footprint fits comfortably within 10-12GB.[33]
3. **Whisper-large-v3:**
   - **FP16:** ~2.87GB VRAM.
   - **INT8/INT4:** Can be compressed to **<1GB** with minimal accuracy loss.[35]
4. **SigLIP 2 (ViT-L/So400m):**
   - Requires ~1-2GB VRAM depending on the variant (Large vs. So400m).[36]

8.2. Total System Budget

- **Minimum Viable Specification (Consumer):**

    - **GPU:** NVIDIA RTX 3090 / 4090 (24GB VRAM).
    - **Configuration:** Qwen2.5-7B (4-bit) + SAM 3 + Whisper (INT8).
    - **Total VRAM Estimate:** ~6GB (LLM) + ~4GB (SAM) + ~1GB (Whisper) + ~2GB (SigLIP) + ~4GB (Context/Cache) = **~17GB**. This leaves headroom for the operating system and display overhead.
- **Recommended Specification (Prosumer/Server):**
    - **GPU:** 2x RTX 4090 or 1x RTX 6000 Ada (48GB).
    - **Configuration:** Qwen2.5-7B (FP16) or Qwen2.5-72B (4-bit) for superior reasoning.
- **NPU Offloading:** Emerging hardware like Ryzen AI NPUs can potentially offload the Whisper transcription task, saving ~1-2GB of VRAM on the main GPU and reducing thermal load.[37]

---

9. Deployment Architecture: The Perception-Reasoning Loop

The final proposed architecture for the project is a **Perception-Reasoning Loop** that orchestrates these models into a cohesive agent.

1. **Input Ingestion:**
    - Video stream enters a ring buffer.
    - Audio stream is separated and sent to **Qwen2-Audio (Whisper)**.
2. **Primary Scan (Visual):**
    - **SAM 3** (Detector Mode) scans keyframes for salient concepts based on the user's current intent (e.g., "watch for packages").
    - **PaddleOCR** runs on defined Regions of Interest (ROI) to extract text (e.g., shipping labels).
3. **Temporal Context:**
    - **InternVideo2.5** (HiCo) maintains a rolling compressed context of the last 5-10 minutes of footage.
4. **Trigger Event:**
    - A "Trigger" (e.g., SAM 3 detects "box", Qwen2-Audio detects "doorbell") activates the **Reasoning Core**.
5. **Reasoning & Response:**
    - **Qwen2.5-VL** receives the current frame (Dynamic Resolution), the compressed temporal context (from InternVideo), the audio transcript, and the OCR text.
    - The Agent generates a response: "A package was just delivered. I verified the label reads 'Priority Mail' and confirmed the delivery truck departed at 10:05 AM."

9.1. Conclusion

This architecture leverages the specific strengths of 2025's SOTA models: **SAM 3** for concept precision, **SigLIP 2** for semantic depth, **InternVideo2.5** for temporal memory, and **Qwen2.5** for agentic reasoning. By strictly adhering to the optimization strategies outlined—specifically

HiCo compression, **sigmoid-loss embedding**, and **4-bit quantization**—this system can be deployed on high-end consumer hardware while delivering performance that rivals massive proprietary clusters. The "missing link" of previous systems—the inability to reliably link a user's verbal concept to a specific pixel mask in a long video—is effectively solved by the SAM 3 + InternVideo2.5 + Qwen2.5 triad.

10. Research References (Integrated)

[1] Ultralytics SAM 3 Documentation; [6] SigLIP 2 Analysis; [5] MarkTechPost SigLIP 2 Review; [2] Ultralytics SAM 3 Blog; [3] LearnOpenCV SAM 3 Deep Dive; [12] InternVideo2 vs VideoMAE Benchmarks; [16] InternVideo2 Technical Report; [10] ModelScope InternVideo2.5; [8] Region Encoder Network (REN); [13] VideoChat-Flash/HiCo; [11] InternVideo2.5 Arxiv; [21] SALMONN-Guard Safety; [19] AlphaXiv Qwen2-Audio; [20] Qwen2-Audio Technical Report; [25] OCR Comparative Study; [22] IntuitionLabs PaddleOCR; [28] InternVideo2 Data Formats; [18] Qwen2.5-VL Technical Report; [29] Video-LLaVA Datasets; [38] VRAM Usage Analysis; [33] APXML Qwen2.5 VRAM Guide; [31] Unsloth Fine-tuning Guide.

Works cited

1. SAM 3: Segment Anything with Concepts - Ultralytics YOLO Docs, accessed December 3, 2025, https://docs.ultralytics.com/models/sam-3/
2. Exploring SAM 3: Meta AI's new Segment Anything Model - Ultralytics, accessed December 3, 2025, https://www.ultralytics.com/blog/exploring-sam-3-meta-ais-new-segment-anything-model
3. SAM-3: What's New, How It Works, and Why It Matters | - Learn OpenCV, accessed December 3, 2025, https://learnopencv.com/sam-3-whats-new/
4. SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features - Hugging Face, accessed December 3, 2025, https://huggingface.co/papers/2502.14786
5. Google DeepMind Research Releases SigLIP2: A Family of New Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features - MarkTechPost, accessed December 3, 2025, https://www.marktechpost.com/2025/02/21/google-deepmind-research-releases-siglip2-a-family-of-new-multilingual-vision-language-encoders-with-improved-semantic-understanding-localization-and-dense-features/
6. Papers Explained 320: SigLIP 2 - Ritvik Rastogi, accessed December 3, 2025, https://ritvik19.medium.com/papers-explained-320-siglip-2-dba08ff09559
7. SigLIP 2: A better multilingual vision language encoder - Hugging Face, accessed December 3, 2025, https://huggingface.co/blog/siglip2
8. REN: Fast and Efficient Region Encodings from Patch-Based Image Encoders - arXiv, accessed December 3, 2025, https://arxiv.org/html/2505.18153v2
9. SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features - ChatPaper, accessed

December 3, 2025, https://chatpaper.com/paper/109859
10. InternVL2.5_HiCo_R64 - Model Details, accessed December 3, 2025, https://modelscope.cn/models/OpenGVLab/InternVL_2_5_HiCo_R64
11. arXiv:2501.00574v3 [cs.CV] 9 Mar 2025, accessed December 3, 2025, https://arxiv.org/pdf/2501.00574?
12. LV-MAE: Learning Long Video Representations through Masked-Embedding Autoencoders - CVF Open Access, accessed December 3, 2025, https://openaccess.thecvf.com/content/ICCV2025/papers/Naiman_LV-MAE_Learning_Long_Video_Representations_through_Masked-Embedding_Autoencoders_ICCV_2025_paper.pdf
13. VideoChat-Flash: Hierarchical Compression for Long-Context Video Modeling, accessed December 3, 2025, https://www.researchgate.net/publication/387671009_VideoChat-Flash_Hierarchical_Compression_for_Long-Context_Video_Modeling
14. InternVideo2.5, The Model That Sees Smarter in Long Videos - DigiAlps LTD, accessed December 3, 2025, https://digialps.com/internvideo2-5-the-model-that-sees-smarter-in-long-videos/
15. OpenGVLab/InternVL_2_5_HiCo_R16 - Hugging Face, accessed December 3, 2025, https://huggingface.co/OpenGVLab/InternVL_2_5_HiCo_R16
16. InternVideo2: Scaling Foundation Models for Multimodal Video Understanding - arXiv, accessed December 3, 2025, https://arxiv.org/html/2403.15377v2
17. InternVideo2: Scaling Foundation Models for Multimodal Video Understanding - arXiv, accessed December 3, 2025, https://arxiv.org/html/2403.15377v3
18. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution, accessed December 3, 2025, https://arxiv.org/html/2409.12191v1
19. Qwen2-Audio Technical Report - alphaXiv, accessed December 3, 2025, https://www.alphaxiv.org/overview/2407.10759
20. Qwen2-Audio Technical Report - arXiv, accessed December 3, 2025, https://arxiv.org/html/2407.10759v1
21. Speech-Audio Compositional Attacks on Multimodal LLMs and Their Mitigation with SALMONN-Guard - arXiv, accessed December 3, 2025, https://arxiv.org/html/2511.10222v1
22. Technical Analysis of Modern Non-LLM OCR Engines | IntuitionLabs, accessed December 3, 2025, https://intuitionlabs.ai/articles/non-llm-ocr-technologies
23. Comparative Performance and Resource Utilization Analysis of OCR Models for Number Plate Recognition on Raspberry Pi 4 - IEEE Xplore, accessed December 3, 2025, https://ieeexplore.ieee.org/iel8/11033710/11034386/11034455.pdf
24. Popular open-source OCR models and how they work - Ultralytics, accessed December 3, 2025, https://www.ultralytics.com/blog/popular-open-source-ocr-models-and-how-they-work
25. Evaluating OCR performance on food packaging labels in South Africa - arXiv, accessed December 3, 2025, https://arxiv.org/html/2510.03570v1
26. A Researcher's Deep Dive: Comparing Top OCR Frameworks | by Aditya Mangal -

Medium, accessed December 3, 2025, https://adityamangal98.medium.com/a-researchers-deep-dive-comparing-top-ocr-frameworks-ca6327b3cc86

27. MMOCR: A Comprehensive Toolbox for Text Detection, Recognition and Understanding, accessed December 3, 2025, https://www.researchgate.net/publication/355384288_MMOCR_A_Comprehensive_Toolbox_for_Text_Detection_Recognition_and_Understanding
28. OpenGVLab/InternVideo2_Vid_Text · Datasets at Hugging Face, accessed December 3, 2025, https://huggingface.co/datasets/OpenGVLab/InternVideo2_Vid_Text
29. Video Instruction Tuning with Synthetic Data - arXiv, accessed December 3, 2025, https://arxiv.org/html/2410.02713v2
30. facebook/sam3 - Hugging Face, accessed December 3, 2025, https://huggingface.co/facebook/sam3
31. unsloth/Qwen2.5-7B-Instruct-1M-bnb-4bit - Hugging Face, accessed December 3, 2025, https://huggingface.co/unsloth/Qwen2.5-7B-Instruct-1M-bnb-4bit
32. VRAM Requirements for LLMs: How Much Do You Really Need? - Hyperstack, accessed December 3, 2025, https://www.hyperstack.cloud/blog/case-study/how-much-vram-do-you-need-for-llms
33. Qwen2.5-7B: Specifications and GPU VRAM Requirements - ApX Machine Learning, accessed December 3, 2025, https://apxml.com/models/qwen2-5-7b
34. GPU System Requirements Guide for Qwen LLM Models (All Variants), accessed December 3, 2025, https://apxml.com/posts/gpu-system-requirements-qwen-models
35. openai/whisper-large-v3 · [AUTOMATED] Model Memory Requirements - Hugging Face, accessed December 3, 2025, https://huggingface.co/openai/whisper-large-v3/discussions/83
36. SigLIP2 - Hugging Face, accessed December 3, 2025, https://huggingface.co/docs/transformers/v4.51.3/model_doc/siglip2
37. Running whisper-large-v3-turbo (OpenAI) Exclusively on AMD Ryzen™ AI NPU - Reddit, accessed December 3, 2025, https://www.reddit.com/r/LocalLLaMA/comments/1odavba/running_whisperlargev3turbo_openai_exclusively_on/
38. How Much VRAM Do You Really Need to Run an LLM? | by Mehmet Hilmi Emel - Medium, accessed December 3, 2025, https://medium.com/@mehmethilmi81/how-much-vram-do-you-really-need-to-run-an-llm-db28f3a79533