Multimodal Fusion with Vision-Language-Action Models for Robotic Manipulation: A Systematic Review

Muhayy Ud Din[a], Waseem Akram[a], Lyes Saad Saoud[a], Jan Rosell[b] and Irfan Hussain*[a]

[a] *Khalifa University Center for Autonomous Robotic Systems (KUCARS) Khalifa University United Arab Emirates.*

[b] *Institute of Industrial and Control Engineering (IOC) Universitat Politecnica de Catalunya Spain.*

**corresponding author: irfan.hussain@ku.ac.ae*

ARTICLE INFO

Keywords:
Information fusion
Vision-Language-Action models
Robotic and embodied control
Foundation models
Language-conditioned manipulation
Robotic manipulation
Multimodal learning
Cross-modal alignment
Embodied AI
Agentic AI

Abstract

Vision Language Action (VLA) models represent a new frontier in robotics by unifying perception, reasoning, and control within a single multimodal learning framework. By integrating visual, linguistic, and action modalities, they enable multimodal fusion systems designed for instruction-driven manipulation and generalist autonomy. This systematic review synthesizes the state of the art in VLA research with an emphasis on architectures, algorithms, and applications relevant to robotic manipulation. We examine 102 models, 26 foundational datasets, and 12 simulation platforms, categorizing them according to their fusion strategies and integration mechanisms. Foundational datasets are evaluated using a novel criterion based on task complexity, modality richness, and dataset scale, allowing a comparative analysis of their suitability for generalist policy learning. We further introduce a structured taxonomy of fusion hierarchies and encoder-decoder families, together with a two-dimensional dataset characterization framework and a meta-analytic benchmarking protocol that quantitatively links design variables to empirical performance across benchmarks. Our analysis shows that hierarchical and late fusion architectures achieve the highest manipulation success and generalization, confirming the benefit of multi-level cross-modal integration. Diffusion-based decoders demonstrate superior cross-domain transfer and robustness compared to autoregressive heads. Dataset analysis highlights a persistent lack of benchmarks that combine high-complexity, multimodal, and long-horizon tasks, while existing simulators offer limited multimodal synchronization and real-to-sim consistency. To address these gaps, we propose the VLA Fusion Evaluation Benchmark to quantify fusion efficiency and alignment. Drawing on both academic and industrial advances, the review outlines future research directions in adaptive and modular fusion architectures, computational resource optimization, and the deployment of interpretable, resource-efficient robotic systems. We further propose a forward-looking agentic VLA paradigm where LLM planners integrate VLA skills as verifiable tools within a closed feedback loop for adaptive and self-improving robotic control. This work provides both a conceptual foundation and a quantitative roadmap for advancing embodied intelligence through multimodal information fusion across robotic domains. A public repository summarizing models, datasets, and simulators is available at: `https://muhayyuddin.github.io/VLAs/`.

Contents

ORCID(s):

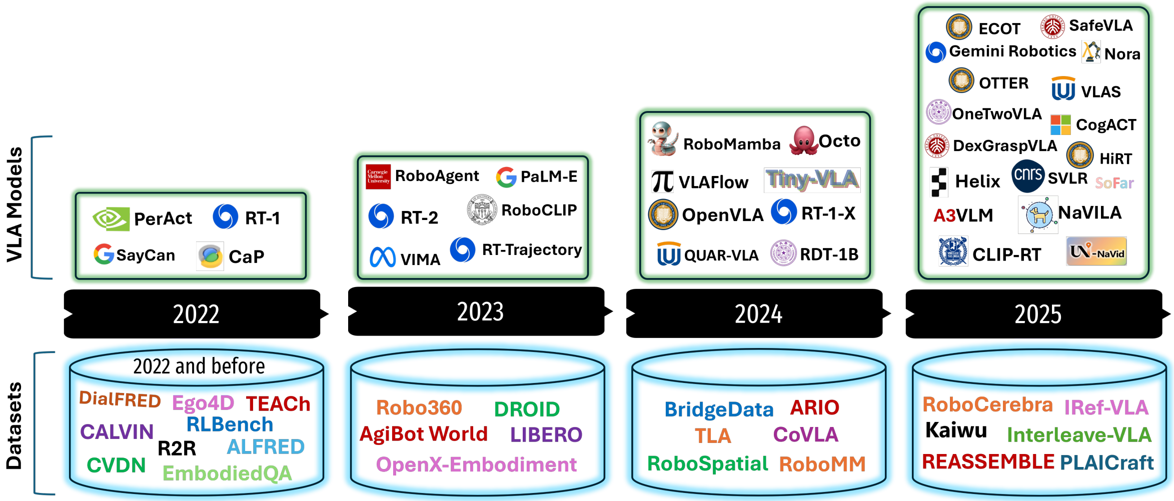


Figure 1: Timeline of Vision-Language-Action (VLA) developments between 2022 and 2025. The top row presents representative VLA models introduced each year, with their associated institutions indicated by logos within red boxes. The bottom row shows major datasets used for training and evaluation, organized by release year. The figure illustrates the increasing scale and diversity of multimodal resources, as well as the growing participation of both academic institutions (e.g., CMU, CNRS, UC, Peking University) and industrial research labs (e.g., Google, NVIDIA, Microsoft). This progression reflects the rapid advancement of multimodal information fusion in robotic manipulation.

1. Introduction

The integration of vision, language, and action into a unified framework has emerged as a paradigm-shifting approach in robotics and embodied artificial intelligence. Traditionally, robotic systems that depend on task-specific programming struggle in dynamic and unstructured environments. In contrast, VLA models aim to utilize the generalization capabilities of large-scale foundation models to enable robotic systems that can understand instructions in natural language, perceive their surroundings, and perform complex tasks autonomously (Brohan et al., 2022; Ahn et al., 2022; Jiang et al., 2022; Team et al., 2024; Din et al., 2025). The fundamental concept of this paradigm is transformer architecture, which has transformed natural language processing and vision with self-attention mechanisms and large-scale pretraining. Models like GPT (Brown et al., 2020), BERT (Devlin et al., 2018), ViT (Dosovitskiy et al., 2020), and CLIP (Radford et al., 2021) demonstrate that massive datasets and parameter scales can produce remarkable generalizability and robustness. These insights have led to new architectures that fuse vision and language into robotics control policies, which step forward towards generalist agents such as RT-1 (Brohan et al., 2022), SayCan (Ahn et al., 2022), RoboRefer (Zhou et al., 2025a), VIMA (Jiang et al., 2022), and Octo (Team et al., 2024). The trend of developing such models is continuously growing as depicted in Fig 1.

The development of effective VLA models is fundamentally dependent on the availability of large-scale, diverse, and multi-model datasets, together with realistic simulation platforms. These elements are essential for training models that can robustly understand language instructions, perceive visual environments, and generate meaningful action sequences. For instance, the *Open X-Embodiment* (Collaboration et al., 2025) dataset unifies data from 22 robot embodiments and more than 500 tasks using a shared action space. It enables the pre-training of foundation models like RT-1-X, significantly enhancing cross-robot generalization. Similarly, the *DROID* dataset (Khazatsky, 2024) uses internet-scale data, combining human-annotated language with robotic video demonstrations for scenes with complex manipulations. These datasets significantly advance the data ecosystem for training and benchmarking VLAs. Comprehensive datasets enable the learning of diverse tasks in both household and industrial contexts. These datasets offer rich annotations, including demonstration trajectories, object state transitions, and diverse natural language prompts. However, real-world data collection is labor-intensive, expensive, and limited in diversity, which highlights the importance of simulation.

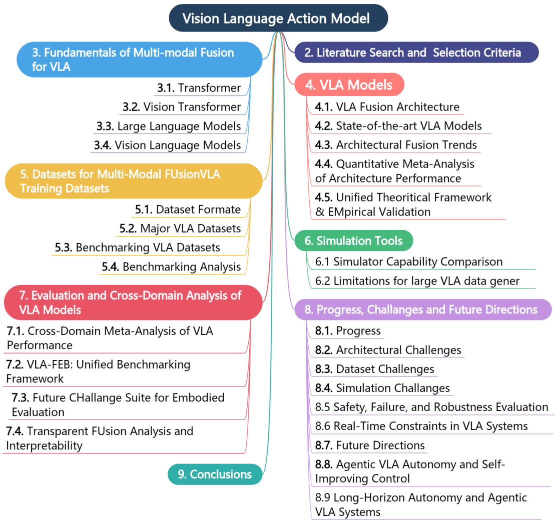
Simulation environments allow data generation to be scaled across a wide range of settings, object types, lighting conditions, and agent embodiments. Platforms such as *Habitat* (Savva et al., 2019), *Isaac Gym* (Makoviychuk et al., 2021), and *RoboSuite* (Zhu et al., 2020) offer programmable and photorealistic environments with physics-based interactions, facilitating both imitation and reinforcement learning paradigms. More recently, tools such as *iGibson* (Xia et al., 2020) and *AI2-THOR* (Kolve et al., 2017) have added support for human-centric indoor environments with naturalistic object arrangements, enhancing semantic realism. Simulation also enables automatic generation of multimodal annotations, such as action trajectories, object states, and natural language instructions that are crucial to align visual, linguistic, and motor modalities. Recent efforts also emphasize the importance of synthetic language generation aligned with task semantics (e.g., VLN-CE (Krantz et al., 2020), ALFRED (Shridhar et al., 2020)) to ensure linguistic diversity and instruction complexity. The integration of simulation and large-scale synthetic datasets is therefore crucial for building VLA systems that are robust, scalable, and applicable to real-world deployment.

Since the field matures at a rapid pace, numerous architectures, datasets, and frameworks are being proposed in a fast order. Despite the growing body of research, there remains a gap in the literature for a comprehensive and systematic synthesis that organizes and categorizes the architectural foundations, benchmark datasets, simulation platforms, and evaluation protocols that collectively shape the current VLA landscape. It is critically needed to contextualize the state-of-the-art and identify emerging patterns, limitations, and opportunities. This study will serve as both a technical reference and a conceptual roadmap to accelerate research in embodied foundation models and generalist robotic intelligence.

There are recent surveys that collectively describe VLA models as advancing from proof-of-concept prototypes toward deployable, general-purpose robotic systems, while emphasizing persistent challenges in fusion design, dataset coverage, and evaluation consistency (Sapkota et al., 2025; Kawaharazuka et al., 2025; Zhong et al., 2025a). Prior reviews have focused on system-level pipelines, architectural trends, and application-oriented roadmaps, highlighting progress in parameter-efficient adaptation, fusion depth, and action tokenization, as well as open problems in long-horizon reasoning, sim-to-real transfer, and safety benchmarking. Building upon these foundations, our work contributes a deeper analytical synthesis by introducing a structured fusion taxonomy, a dataset complexity framework, and a quantitative meta-analysis that links architectural choices, fusion hierarchy, encoder scale, decoder class, and multimodal coverage, to empirical performance across 102 VLA models. In contrast to prior descriptive surveys, this article establishes statistical and theoretical connections between design variables and success metrics, offering an integrated view of how fusion dynamics, generalization, and safety co-evolve in embodied intelligence.

Contributions. This work offers a comprehensive synthesis and analytical framework for understanding and evaluating VLA models for robotics. Building on an extensive survey of models, datasets, and simulators, the major contributions of this review are as follows:

1. *Unified Taxonomy of VLA Architectures.* We present a structured and hierarchical taxonomy that organizes over one hundred VLA models according to their fusion hierarchy, encoder scale, action decoder design, and multimodal integration strategy. This taxonomy consolidates diverse architectural patterns and provides a unified perspective for understanding emerging VLA design trends.



Vision Language Action Model
3. Fundamentals of Multi-modal Fusion for VLA
3.1. Transformer
3.2. Vision Transformer
3.3. Large Language Models
3.4. Vision Language Models
5. Datasets for Multi-Modal FUsionVLA Training Datasets
5.1. Dataset Formate
5.2. Major VLA Datasets
5.3. Benchmarking VLA Datasets
5.4. Benchmarking Analysis
7. Evaluation and Cross-Domain Analysis of VLA Models
7.1. Cross-Domain Meta-Analysis of VLA Performance
7.2. VLA-FEB: Unified Benchmarking Framework
7.3. Future CHallange Suite for Embodied Evaluation
7.4. Transparent FUsion Analysis and Interpretability
9. Conclusions
2. Literature Search and Selection Criteria
4. VLA Models
4.1. VLA Fusion Architecture
4.2. State-of-the-art VLA Models
4.3. Architectural Fusion Trends
4.4. Quantitative Meta-Analysis of Architecture Performance
4.5. Unified Theoritical Framework & EMpirical Validation
6. Simulation Tools
6.1 Simulator Capability Comparison
6.2 Limitations for large VLA data gener
8. Progress, Challanges and Future Directions
8.1. Progress
8.2. Architectural Challenges
8.3. Dataset Challenges
8.4. Simulation Challanges
8.5 Safety, Failure, and Robustness Evaluation
8.6 Real-Time Constraints in VLA Systems
8.7. Future Directions
8.8. Agentic VLA Autonomy and Self-Improving Control
8.9 Long-Horizon Autonomy and Agentic VLA Systems

Figure 2: Overview of the skeleton of the paper, highlighting the main sections and their interrelated subtopics.

2. *Dataset Complexity Framework and Simulation Ecosystem Analysis.* We introduce a two-dimensional characterization of dataset difficulty using task complexity (C_{task}) and modality richness (C_{mod}), mapping 26 foundational datasets within this space. In parallel, we survey 12 major simulation platforms, evaluating their multimodal data-generation accuracy, embodiment realism, and support for sim-to-real transfer, providing a clearer understanding of the data and simulation landscape.
3. *Large-Scale Quantitative Meta-Analysis.* We conduct a statistical meta-analysis across 102 VLA models, linking core design variables including fusion depth, encoder/decoder families, and dataset attributes to standardized measures of manipulation success and generalization. This analysis offers the first quantitative evidence on how architectural and dataset choices shape empirical VLA performance.
4. *Unified Benchmarking and Research Roadmap.* We propose the *Vision Language Action Fusion Evaluation Benchmark* (VLA-FEB), introducing metrics such as the Cross-Modal Alignment Score (CMAS) and Fusion Energy Index (FEI) to systematically assess fusion efficiency and cross-modal coherence. Building on the empirical insights, we outline future research directions spanning adaptive multimodal fusion, efficient tokenization, standardized dataset protocols, and improved sim-to-real integration pipelines, establishing a roadmap for advancing embodied multimodal intelligence.

The remainder of the paper is organized as illustrated in Fig. 2. Sec. 2 describes the literature search strategy and selection criteria used to identify the most relevant and representative VLA models, datasets, and simulation platforms. Sec. 3 introduces the fundamentals of multimodal fusion, outlining the underlying architectures of transformers, vision transformers, large language models, and vision-language models that collectively form the basis of modern VLA systems. Sec. 4 presents a comprehensive review of VLA architectures, highlighting state-of-the-art systems, fusion hierarchies, and architectural trends, followed by a quantitative meta-analysis and a unified theoretical framework linking design variables to performance outcomes. Sec. 5 analyzes the major multimodal datasets used for training and evaluating VLA systems, introducing a task-modality characterization framework to identify underexplored regions in the dataset landscape. Sec. 6 reviews the simulation environments and tools supporting large-scale multimodal data generation, embodied learning, and sim-to-real transfer. Sec. 7 introduces the unified benchmarking protocol (VLA-FEB) for quantitative evaluation, presents cross-domain meta-analysis findings, and outlines the proposed Future Challenge Suite for embodied assessment. Sec. 8 discusses key challenges and future research directions, categorizing them under architectural, dataset, and simulation aspects, and highlights emerging solutions such as adaptive fusion modules, differentiable simulators, and standardized multimodal capture pipelines. Finally, Sec. 9 concludes the review and provides a roadmap for advancing generalist robotic autonomy through robust multimodal fusion and embodied intelligence.

2. Literature Search and Selection Criteria

We conducted an extensive search through IEEE Xplore, Elsevier, Springer Nature, MDPI, Wiley, and arXiv to identify work on VLA models, VLA datasets, and simulation tools for robotic simulation and data generation. To capture each aspect, we developed sets of keywords; for *VLA models*: “vision language action” OR VLA OR (vision AND language AND action) OR “Vision Language Models for Robotic manipulation”, “Multimodal Robotic Control”, “Vision-Language Grounding”, “Visuomotor Transformer”, “End-to-End Robot Learning”. The keywords for *Training datasets* are “VLA dataset” OR “manipulation dataset” OR “embodied AI dataset for manipulation”. Finally for *Simulation tools*: “simulator for robotic manipulation” OR “embodied AI data-generation simulator” OR “robotic manipulation simulator” OR “realistic dynamic simulation for robotic manipulation” We applied the following inclusion criteria to identify original research: 1) Proposes or evaluates a VLA model, a VLA dataset, or a simulator for robotic manipulation or manipulation data generation. 2) Presents a novel model, dataset, or a new simulator.

To ensure comprehensive coverage, we complement traditional database searches with conversational queries in a large-language model (e.g. GPT), using targeted prompts per thematic area. For *VLA models*, we asked GPT to "List vision-language-action models published between 2022 and 2025," "List end-to-end transformer-based VLA architectures", "List VLA methods that use diffusion-based action decoders", "List modular fusion framework VLA models". For datasets, prompts included "List of VLA datasets released between 2022 and 2025", "List manipulation datasets used for VLM training" and "List embodied AI datasets for robotics" and "List of well-known manipulation datasets". For simulation tools, we queried "List simulators for robotic manipulation data generation," "List simulation environments for embodied AI dataset creation," "List robotic manipulation dataset generation simulators" and "List robotic dynamic simulators". We then merged these lists with our database results, removed duplicates, and performed manual validation to arrive at the final set of models, datasets, and simulators included in this review.

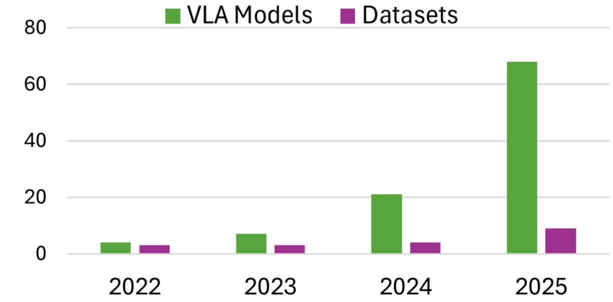


Figure 3: Annual VLA models and foundational VLA datasets count from 2022 to 2025. Green bars indicate the number of new VLA model introduced each year, while purple bars represent the number of novel dataset releases. The data illustrate a rapid acceleration in model development, particularly in 2025, alongside steady growth in dataset creation to support training and evaluation of these models.

We also included arXiv e-prints (`https://arxiv.org/`) in our search because the field of VLAs has recently begun to mature, and most breakthroughs and novel architectures have appeared as preprints in recent times. Fig. 3 shows the VLA (in green) and dataset (in purple) counts per year used in this work. We thoroughly examined the preprints and included only those that make substantial contributions to the field. The integration of preprints ensures that we capture the very latest models, datasets, and simulation tools as soon as they emerge, giving a more accurate and up-to-date picture of this rapidly evolving field.

3. Fundamentals of Multi-Modal Fusion for VLA

This section outlines the core architectures behind VLA models. We start with the transformer, a model that has transformed both language and vision tasks. We then cover Vision Transformers (ViTs), which apply self-attention to image patches for visual feature extraction. The subsequent category comprises Large Language Models (LLMs), which are transformer-based models trained on large text datasets that perform reasoning, instruction following, and zero-shot tasks. Finally, we will provide an overview of vision language models (VLMs), which fuse visual and textual data through cross-modal attention to ground instructions in robotic actions. These components form the backbone of modern VLA architectures, detailed in the following subsections.

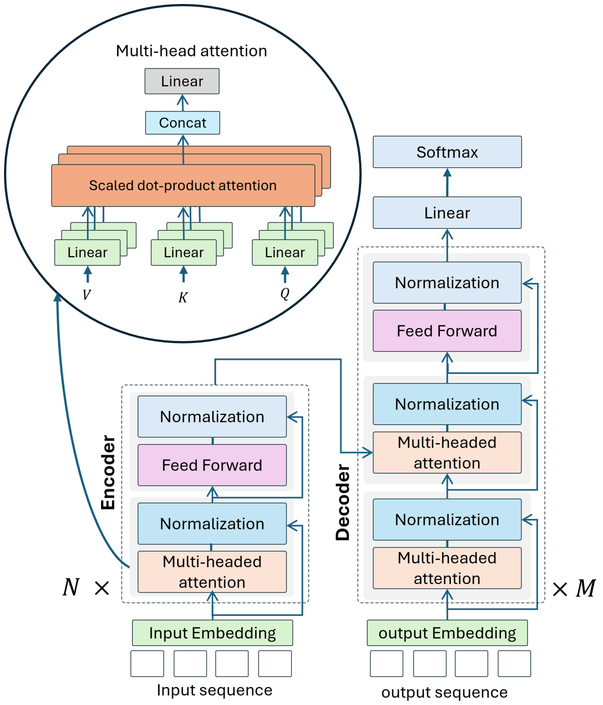


Figure 4: An overview of the Transformer architecture highlighting the encoder-decoder structure and the internal mechanism of multi-head attention. The encoder processes input embeddings through layers of multi-head attention, normalization, and feedforward networks. The decoder mirrors this with additional masked attention layers and incorporates encoder outputs for contextual decoding. The magnified view illustrates the scaled dot-product attention and how multiple attention heads are concatenated and linearly transformed to form the final multi-head attention output. The image is adapted from (Vaswani et al., 2017)

3.1. Transformer

Transformers are a class of deep learning architectures introduced by Vaswani et al. (Vaswani et al., 2017), these models have revolutionized the fields of natural language processing and computer vision by enabling greater parallelization and scalability in sequence modeling, i.e. in the processing and learning of patterns from sequences of data. The core of Transformers lie in the *self-attention mechanism*, detailed below, which let each token in a sequence weigh and combine information from all other tokens to build a context-aware representation.

The Transformer architecture (depicted in Fig 4) comprises three differentiated parts: the embedding layer, the encoder stack, and the decoder stack, which includes the final output projection and softmax layer. They are detailed in the following subsections, after presenting the *self-attention mechanism*.

3.1.1. Self-Attention

The self-attention mechanism allows each token in a sequence to attend to all other tokens when computing its representation. To achieve this, the model first transforms each input token into three distinct vector representations: queries (Q), keys (K), and values (V). These vectors are obtained through learned linear projections of the input embeddings and serve different roles in the attention computation: the *query* represents what the current token is looking for, the *key* indicates what information each token provides, and the *value* contains the actual content to be shared. By comparing queries to keys, the model determines attention weights, which are then used to aggregate the values into a new context-aware representation for each token.

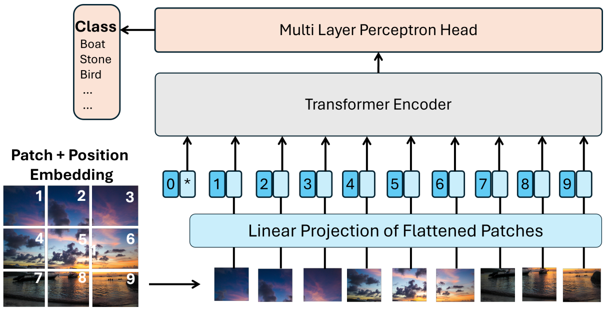

Class
Boat
Stone
Bird
...
...
Multi Layer Perceptron Head
Transformer Encoder
Patch + Position Embedding
0
*
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches
1
2
3
4
5
6
7
8
9

Figure 5: Architecture of the ViT. The input image is divided into fixed-size non-overlapping patches which are flattened and linearly projected into embedding vectors. A learnable classification (CLS) token is prepended to the sequence of patch embeddings (shown in darker blue). Positional embeddings are added to retain spatial information before feeding the sequence into a standard Transformer encoder. The output of the CLS token is passed through an MLP head to produce the final class prediction. The image is adpated from (Dosovitskiy et al., 2021)

This process is formalized by the *scaled dot-product attention* mechanism, which computes the dot product between queries and keys, scales the result by $\sqrt{d_k}$ (where d_k is the dimension of the keys), applies a softmax function to obtain attention weights, and finally combines these weights with the values to produce the output:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{1}$$

Building upon this mechanism, *multi-head attention* extends the model's capacity by enabling it to attend to information from multiple representation subspaces simultaneously. Instead of computing attention once with a single set of projection matrices, the model uses h parallel attention layers, or *heads*, each with its own learned weight matrices W_i^Q, W_i^K, and W_i^V. These matrices independently project the input into h different subspaces, allowing each head to focus on different aspects of the input relationships:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{2}$$

$$\text{where } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

The outputs from all heads are concatenated and passed through a final linear projection using W^O to produce the result of the multi-head attention module.

3.1.2. Embeddings

Embedding refers to continuous vector representations that map discrete inputs such as words, image patches, or action tokens into a dense latent space. In Transformer architectures, embeddings serve as the foundational interface between symbolic input (for example, textual instructions) and differentiable computation. Input embeddings enable the model to process structured data in a unified format, while output embeddings decode latent representations into action or token spaces. Since the Transformer architecture lacks recurrence, a set of sinusoidal positional encodings is added to the input embeddings to provide information about the order of elements in a sequence. These learned representations capture semantic, spatial, or temporal relationships that are essential for generalization and reasoning in multimodal tasks (Vaswani et al., 2017; Dosovitskiy et al., 2021; Press and Wolf, 2017).

3.1.3. Encoder

The encoder stack consists of N identical layers, where each layer has a multi-head self-attention mechanism followed by a position-wise feedforward network. Both sublayers are equipped with residual connections (which add the input of a sub-layer to its output to help preserve information and ease optimization) and layer normalization (which normalizes activations to improve training stability). In the first layer, the input embeddings provide the queries, keys, and values. In deeper layers, these are derived from the output of the preceding layer.

3.1.4. Decoder

The decoder stack also consists of M identical layers, but with a slightly different architecture. Each layer contains three sub-layers: a masked multi-head self-attention layer, a multi-head encoder-decoder attention layer, and a feedforward network. The masked self-attention prevents a position from attending to subsequent tokens, ensuring that the model generates output in an autoregressive manner. The encoder-decoder attention enables the decoder to query the encoder's outputs, integrating source information into the generation process. Like the encoder, each sub-layer in the decoder is followed by residual connections and layer normalization.

The final output of the decoder is passed through a linear transformation and a softmax function to produce a probability distribution over all possible output tokens, enabling the model to predict the next token in the target sequence one step at a time.

3.2. Vision Transformers

The Vision Transformer extends the Transformer architecture to visual domains by treating image patches as input tokens (Dosovitskiy et al., 2020). As shown in Fig 5, an image is split into a sequence of non-overlapping patches, each of which is linearly projected into a fixed-dimensional embedding. These embeddings are then augmented with learnable positional encodings and passed through a standard Transformer encoder. A classification token is appended, and its final representation is used for prediction through a multi-layer perceptron head (MLP). ViTs achieve competitive performance on benchmarks like ImageNet (Deng et al., 2009), highlighting the power of attention in learning global visual representations.

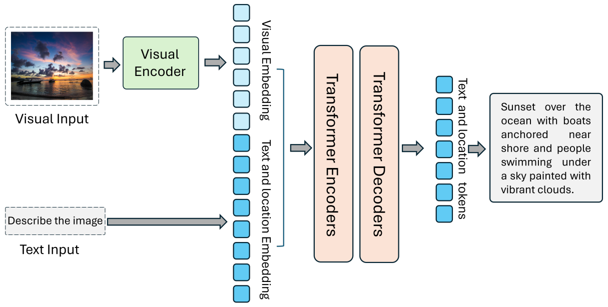

Visual Input
Visual Encoder
Visual Embedding
Text and location Embedding
Describe the image
Text Input
Transformer Encoders
Transformer Decoders
Text and location tokens
Sunset over the ocean with boats anchored near shore and people swimming under a sky painted with vibrant clouds.

Figure 6: Architecture of a VLM for image captioning and semantic understanding. Visual input is processed by a visual encoder to extract patch-based embeddings. In parallel, a text prompt (e.g., "Describe the image") is tokenized and embedded. These embeddings are fused and jointly processed through a Transformer-based encoder-decoder architecture. The model outputs a natural language caption that describes the semantic content of the image, enabling tasks such as captioning, question answering, and visual grounding. the image is adapted from (Xiao et al., 2024)

3.3. Large Language Models

LLMs are transformer architectures trained on large-scale text data. These models are typically classified into three architectural types: *encoder-only*, *decoder-only*, and *encoder-decoder*. Each structure is optimized for different types of task in natural language processing and robotics applications. *Encoder-only* models, such as BERT, RoBERTa, and DeBERTa, utilize bidirectional self-attention mechanisms to learn deep contextual relationships from both the left and right of a token (Ghojogh et al., 2024; Zayyanu et al., 2024). These models are trained using masked language modeling (MLM) objectives and are well-suited for text classification, semantic similarity, and question answering. Their strength lies in representing sentence-level meaning rather than in generating sequential text.

Decoder-only models, such as GPT-3, GPT-4, PaLM, and LLaMA, operate sequentially by predicting the next token in a sequence based only on previous tokens (Ghojogh et al., 2024). This unidirectional architecture is optimized for text generation, dialogue, summarization, and other open-ended generative tasks.

Encoder-decoder models, also known as sequence-to-sequence architectures, such as T5, BART, and the original Transformer, include an encoder to represent the input and a decoder to produce the output (Liu et al., 2021; Brandisauskas et al., 2023; Ghojogh et al., 2024). These are ideal for tasks where full input processing is required before output generation, including machine translation, summarization, and code generation. Recent applications in robotics also use encoder-decoder models for instruction grounding and language-conditioned action planning.

3.4. Vision Language Models

VLMs use the synergy between computer vision and natural language processing to perform tasks such as image captioning, visual question answering, cross-modal retrieval, and instruction grounding. Like LLMs, they are built on the Transformer architecture, using either dual encoders (e.g., CLIP), unified encoder-decoder frameworks (e.g., BLIP, Flamingo), or sequence-to-sequence stacks. During training, they typically align visual and textual inputs using objectives such as contrastive learning (matching image-text pairs), masked modeling (predicting masked tokens or regions), or learning to generate captions from images.

Fig. 6 illustrates a typical VLM designed for image captioning using a transformer-based encoder-decoder architecture. The process begins with an input image, which is divided into patches and embedded via a *visual encoder*, such as a (ViT) (Mishra et al., 2024; Lam et al., 2023). A language prompt, e.g., "Describe the image", is also tokenized and embedded, either directly or via lightweight text guidance. The visual tokens and prompt embeddings are passed through a *Transformer encoder-decoder* stack. The encoder captures spatial and semantic relationships from the image, while the decoder generates output tokens autoregressively. This allows the model to produce a fluent and contextually accurate caption, as shown in Fig. 6. This architecture enables semantic-level reasoning over visual input and has shown strong performance on benchmark datasets such as Flickr8k and Flickr30k (Mishra et al., 2024; Abdel-Hamid et al., 2024).

Recent studies have demonstrated improved caption quality through the use of advanced modules or architectural variants within the encoder-decoder stack, such as Swin Transformers, T5-based decoders, or GPT-based language models, depending on the design (Lam et al., 2023; Mishra et al., 2024; Fouad et al., 2024). These approaches are widely used in robotic perception systems where visual scene understanding must be paired with natural language generation or instruction.

4. Vision Language Action Models

VLA models represent a new frontier in robotic intelligence, enabling robots to perceive visual environments, understand natural language instructions, and execute grounded actions accordingly. These models bridge the semantic gap between multimodal inputs, such as images, sensor data, human commands, and low-level robotic control. VLA architectures are particularly relevant for unstructured and dynamic environments, where traditional rule-based programming is infeasible. They empower robots to generalize tasks such as object manipulation, navigation, and interaction by using deep learning, representation alignment, and sequential decision making (Wang et al., 2024a; Guruprasad et al., 2024; Wang et al., 2024d).

4.1. VLA Fusion Architectures

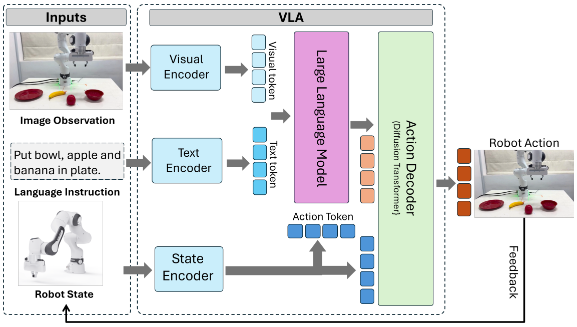


Figure 7: Architecture of a VLA system for robotic manipulation. The model processes three inputs: an image of the scene, a natural language instruction, and the robot's internal state. These are encoded respectively through visual, text, and state encoders. The resulting embeddings are passed to an LLM that fuses multimodal information and generates a semantic representation of the intended task. This representation, along with robot state features, is processed by an Action Decoder implemented as a diffusion transformer to generate a trajectory that accomplishes the commanded task.

The VLA architecture illustrated in Fig. 7 represents an end-to-end framework that is a representative of the leading VLA systems such as; RT-2 (Zitkovich et al., 2023), OpenVLA (Driess et al., 2024), CLIP-RT (Kang et al., 2025), Octo (Team et al., 2024), and RT-1 (Brohan et al., 2022), all of which employ transformer-based vision and language backbones fused through cross-modal attention.

The architecture unifies three parallel encoder streams: visual, linguistic, and proprioceptive, into a single transmission diffusion backbone that directly generates control commands. First, a transformer-based *Visual Encoder* (e.g. ViT (Dosovitskiy et al., 2020), DINOv2 (Caron et al., 2021)) processes raw RGB (depth/semantic) images of the workspace and produces a sequence of fixed-length feature tokens. In parallel, a pre-trained *Language Encoder* (for example, PaLM (Chowdhery et al., 2022) or LLaMA (Touvron et al., 2023)) tokenizes and embeds natural language instructions, whether high-level goals (e.g., "Put bowl, apple, and banana on plate.") or detailed stepwise instructions, in the same d-dimensional space. Similarly, *State Encoder* embeds the proprioceptive and kinematic state of the robot (joint angles, pose of the end effector, gripper status) through an MLP or small transformer into additional tokens, allowing the model to reason about reachability, collision avoidance, and feedback correction.

All tokens are concatenated and passed into a transformer-based model that produces *action embeddings*. This model may implement either a diffusion policy, using a *Diffusion Transformer* that iteratively denoises a noisy latent trajectory (e.g., as in Diffusion Policy (Chi et al., 2023a) or VLAFlow (Black et al., 2024b)), or a direct policy, which predicts the embeddings in a single pass without diffusion. At inference time, the action embeddings are converted into continuous control signals, such as end-effector velocities or joint torques, either through a lightweight output head or by completing the full diffusion sampling process. In some implementations, the embeddings can also be decoded into imagined next-frame images, enabling an "imagine-and-verify" loop for closed-loop execution.

Table 1: Comprehensive survey of VLA models developed for robotic manipulation and instruction-driven autonomy. The table lists each model's name, publication year, indicates whether it operates in an end-to-end fashion (mapping raw visual and language inputs directly to actions) or focuses on individual components (vision, language, or action modules), and summarizes its main technical contributions. The final column details the primary training datasets and core model components-including vision encoders, language encoders, and action decoders used by each method.

Model Name	**End-to-End**	**Comp. Focused**	**Main Contributions**	**Training Dataset and Model Components**
CLIPort (Shridhar et al., 2022a)	✔	✔	Pioneered the semantic grounding of visuomotor policies by integrating CLIP features into dense transport maps for precise pick-and-place.	Dataset: Self-collected visuomotor demos; Vision Encoder: CLIP-ResNet50 + Transporter-ResNet; Language Encoder: CLIP text encoder; Action Decoder: LingUNet
RT-1 (Brohan et al., 2022)	✔	✔	Introduced a discretized action transformer for scalable multi-task kitchen manipulation.	Dataset: Self-collected RT-1-Kitchen; Vision Encoder: EfficientNet CNN; Language Encoder: Universal Sentence Encoder; Action Decoder: Discretized action transformer head
Gato (Reed et al., 2022)	✔	✔	Demonstrated a unified tokenization scheme across vision, language, and control tasks, achieving zero-shot transfer across domains.	Dataset: Self-collected multi-domain tasks; Vision Encoder: custom ViT; Language Encoder: Sentence Piece tokenizer; Action Decoder: Autoregressive Transformer
VIMA (Jiang et al., 2022)	✔	✔	Handled six distinct vision-language grounding tasks via a prompt-based multimodal policy.	Dataset: VIMA self-collected; Vision Encoder: Mask R-CNN; Language Encoder: T5-base; Action Decoder: Transformer policy head
PerAct (Shridhar et al., 2023)	✔	✘	Uses voxel-based representation with language conditioning for high-precision manipulation; operates directly on point cloud voxels.	Dataset: RLBench; Vision Encoder: Perceiver Transformer + voxel grid encoder; Language Encoder: CLIP text encoder; Action Decoder: Transformer voxel policy head
SayCan (Ahn et al., 2022)	✘	✔	Combines language model planning with value function grounding in the real world; interprets high-level goals into executable robot actions.	Dataset: Self-collected everyday manipulation demos; Vision Encoder: none; Language Encoder: PaLM; Action Decoder: Value-conditioned execution module
RoboAgent (Bharadhwaj et al., 2023)	✔	✘	MT-ACT: multi-task transformer policy with semantically augmented CVAE encoding and action-chunking for strong real-world generalization.	Dataset: RoboSet teleop demos; Vision Encoder: Multi-view CNN encoder; Language Encoder: Semantic transformer encoder; Action Decoder: CVAE + Chunked trajectory predictor
RT-Trajectory (Gu et al., 2023)	✔	✘	Conditioned policies on user-sketched trajectories to improve generalization to novel layouts and paths.	Dataset: RT-1 dataset; Vision Encoder: EfficientNet-B3; Language Encoder: None; Action Decoder: Sketch-conditioned behavioral cloning policy
ACT (Zhao et al., 2023)	✔	✔	Applied temporal ensembling to achieve smooth bimanual manipulation with 0.1 mm precision.	Dataset: self-collected demos on ALOHA; Vision Encoder: ResNet-18; Language Encoder: none ; Action Decoder: CVAE-Transformer head
RT-2 (Zitkovich et al., 2023)	✔	✔	First large VLA co-finetuned on Internet VQA and robot data, unlocking emergent multi-robot zero-shot capabilities.	Dataset: Internet VQA + RT-1-Kitchen; Vision Encoder: PaLI-X/PaLM-E ViT; Language Encoder: PaLI-X/PaLM-E text encoder; Action Decoder: Symbol-tuning transformer
VoxPoser (Huang et al., 2023)	✔	✔	Achieved zero-shot constraint-aware motion planning by composing a frozen VLM and LLM without additional training.	Dataset: Self-collected motion demos+RLBench; Vision Encoder: OWL-ViT; Language Encoder: GPT-4; Action Decoder: MPC optimizer
CLIP-RT (Kang et al., 2024)	✘	✔	Contrastive policy using CLIP vision and text encoders to select language-based motion primitives, enabling fast learning and robust zero-shot transfer for table-top manipulation.	Dataset: OXE; Vision Encoder: CLIP image encoder (ViT-H-14); Language Encoder: CLIP text encoder; Action Decoder: Cosine similarity head over text-embedded motion primitives.
Diffusion Policy (Chi et al., 2023a)	✔	✔	Introduced diffusion-based policy modeling for multi-modal visuomotor action distributions.	Dataset: Self-collected demos; Vision Encoder: ResNet-18; Language Encoder: None; Action Decoder: diffusion policy network
Octo (Team et al., 2024)	✔	✔	First generalist diffusion policy trained on 4 M+ trajectories across 22 robot platforms, demonstrating broad transfer.	Dataset: Open X-Embodiment; Vision Encoder: CNN encoder ; Language Encoder: T5-base; Action Decoder: Diffusion Transformer head
VLATest (Wang et al., 2024c)	✘	✔	Automated framework for large-scale VLA model testing, revealing robustness gaps and guiding improvements in manipulation.	Dataset: none ; Vision Encoder: that used in (e.g., OpenVLA,RT-1,Octo); Language Encoder: that used in (e.g., OpenVLA,RT-1,Octo); Action Decoder: that used in (e.g., OpenVLA,RT-1,Octo)
NaVILA (Cheng et al., 2024)	✔	✘	Hierarchical planning yields 88% real-world navigation success for legged robots via language-conditioned topological control.	Dataset: Self-collected Real-world legged robot demos; Vision Encoder: VILA Vision Encoder ; Language Encoder: VILA LLM; Action Decoder: Topological graph planner + RL policy for joint commands
RoboNurse-VLA (Li et al., 2024c)	✔	✘	Real-time voice-to-action pipeline for surgical instrument handover, robust to unseen tools in dynamic scenes.	Dataset: Self-collected Surgical handover videos + voice prompts; Vision Encoder: SAM2 ; Language Encoder: LLaMA-2; Action Decoder: token-based action decoder
Mobility VLA (Chiang et al., 2024)	✔	✘	Multimodal instruction navigation with topological mapping for robust long-range mobility.	Dataset: MINT instruction tours; Vision Encoder: Gemini 1.5 Pro based (ViT); Language Encoder: Gemini 1.5 Pro based text encoder; Action Decoder: Topological graph-based planner
RevLA (Dey et al., 2024)	✔	✘	Domain adaptation adapters to improve the generalization of robotic foundation models across visual domains.	Dataset: Open X-Embodiment (OXE); Vision Encoder: DINO-v2 + SigLIP; Language Encoder: LLama-7B; Action Decoder: Llama head, outputs 7 discrete action tokens
Uni-NaVid (Zhang et al., 2024b)	✔	✘	Video-based VLA unifying embodied navigation tasks across multiple benchmarks.	Dataset: Room-to-Room (R2R) + REVERIE; Vision Encoder: EVA-CLIP; Language Encoder: Vicuna-7B; Action Decoder: Vicuna-7B head (4 discrete action tokens)
RDT-1B (Liu et al., 2024c)	✔	✔	1.2B-parameter diffusion foundation model excelling at bimanual manipulation and zero-shot generalization.	Dataset: self-collected 6K ALOHA episodes; Vision Encoder: SigLIP ; Language Encoder: T5-XXL ; Action Decoder: Diffusion Transformer + MLP decoder
RoboMamba (Liu et al., 2024b)	✔	✘	Mamba-based unified VLA with linear-time inference for real-time robotic reasoning.	Dataset: SAPIEN sim benchmarks + real-world demos; Vision Encoder: Mamba VLM visual backbone; Language Encoder: Mamba VLM text backbone; Action Decoder: MLP policy head for SE(3) pose predicting
Chain-of-Affordance (Li et al., 2024a)	✘	✔	Sequential affordance reasoning for spatial planning, achieving SOTA on LIBERO dataset.	Dataset: LIBERO + real/sim manipulation tasks; Vision Encoder: Qwen2-VL; Language Encoder: Qwen2-VL; Action Decoder: Diffusion policy head
Edge VLA (Budzianowski et al., 2024)	✔	✔	Lightweight, edge-optimized VLA for low-power real-time inference.	Dataset: OXE + Bridge robotics set; Vision Encoder: SigLIP + DINOV2; Language Encoder: Qwen2; Action Decoder: Non-autoregressive control head
OpenVLA (Kim et al., 2024)	✔	✔	LORA-fine-tuned open-source VLA achieving efficient transfer and high success.	Dataset: OXE + DROID robot data; Vision Encoder: DINOv2 + SigLIP; Language Encoder: Llama 2; Action Decoder: Llama 2 output head (predicts discretized action tokens as output)
CogACT (Li et al., 2024b)	✔	✔	Componentized diffusion action transformer, +59.1% success over OpenVLA with specialized adaptation.	Dataset: OXE subset + real trials; Vision Encoder: DINOv2 + SigLIP; Language Encoder: LLaMA-2; Action Decoder: Diffusion Transformer head
ShowUI-2B (Lin et al., 2024)	✔	✔	GUI/web navigation via screenshot grounding and efficient token selection.	Dataset: 256 K GUI instruction demos; Vision Encoder: Qwen2-VL-2B ViT; Language Encoder: Qwen2-VL-2B LLM; Action Decoder: Qwen2-VL-2B output head (GUI actions as tokens)
Pi-0 (Black et al., 2024a)	✔	✘	General robot control flow model for high-frequency, open-world tasks.	Dataset: Extended OXE called Pi-Cross-Embodiment; Vision Encoder: PaliGemma (SigLIP); Language Encoder: PaliGemma (Gemma-2B); Action Decoder: diffusion-based Flow matching action expert head
HiRT (Zhang et al., 2024a)	✔	✔	Hierarchical planning/control separation, doubling execution speed and improving dynamic task success.	Dataset: Self collected Real-world data; Vision Encoder: InstructBLIP; Language Encoder: LLaMA-2; Action Decoder: Latent-conditioned policy head (MLP)
A3VLM (Huang et al., 2024)	✘	✔	Learns articulation-aware affordance grounding purely from RGB video, generalizing to unseen object joints.	Dataset: PartNet-Mobility; Vision Encoder: CLIP, DINOv2, Q-Former (fused); Language Encoder: LLaMA2; Action Decoder: Parameterized primitive motion generator
SVLR (Samson et al., 2024)	✘	✔	Modular "segment-to-action" pipeline using visual prompt retrieval for on-device policy execution.	Dataset: Self-collected visual prompts; Vision Encoder: Mini InternVL; Language Encoder: Phi-3mini4k; Action Decoder: Script-based action binder
Bi-VLA (Gbagbe et al., 2024)	✘	✔	Dual-arm instruction-to-action planner grounded in recipe demonstrations, achieving 83.4 % real-world task success.	Dataset: Visual-recipe demos; Vision Encoder: Qwen-VL; Language Encoder: Starling-LM; Action Decoder: Python trajectory generator

continue

Model Name	End-to-End	Comp. Focused	Main Contribution	Training Dataset and Model Components
QUAR-VLA (Ding et al., 2024)	✔	✘	Quadruped-specific VLA with adaptive gait and body command mapping, strong sim-to-real transfer.	Dataset: QUART locomotion + manipulation; Vision Encoder: EfficientNet-B3; Language Encoder: FiLM / VLM tokenizer; Action Decoder: Transformer decoder (discrete tokens)
3D-VLA (Zhen et al., 2024)	✔	✔	Integrates 3D generative diffusion heads for world reconstruction, enabling planning in RGB+D and point-cloud spaces.	Dataset: 3D-language-action pairs; Vision Encoder: 3D-aware transformer; Language Encoder: 3D-LLM; Action Decoder: Multi-head diffusion planner
RoboMM (Yan et al., 2024)	✔	✔	MIM-based multimodal decoder unifying 3D perception and language for spatially aligned policy fusion.	Dataset: RoboData (CALVIN, Meta-World); Vision Encoder: Multi-view CLIP + occupancy network; Language Encoder: Flamingo-style fusion module; Action Decoder: Multimodal MLP/attention decoder
FAST (Pertsch et al., 2025)	✔	✔	Frequency-space action tokenization for up to 15 times faster inference on general robot control.	Dataset: DROID; Vision Encoder: PaliGemma (SigLIP); Language Encoder: PaliGemma (Gemma-2B); Action Decoder: FAST token generator
OpenVLA-OFT (Kim et al., 2025)	✔	✔	Optimized fine-tuning of OpenVLA with parallel chunked decoding, achieving 97.1 % success on LIBERO dataset and 26 time speed-up.	Dataset: LIBERO; Vision Encoder: SigLIP + DINOv2; Language Encoder: LLaMA-27B; Action Decoder: Llama 2 Parallel chunking head
CoVLA (Arai et al., 2025)	✔	✘	VLA model for autonomous driving, trained on richly annotated scene data for robust planning.	Dataset: Large-scale driving videos + annotations; Vision Encoder: CLIP ViT; Language Encoder: LLaMA-2; Action Decoder: Trajectory prediction module
ORION (Fu et al., 2025)	✔	✘	Holistic end-to-end driving VLA aligning semantic understanding with generative trajectory control.	Dataset: E2E driving benchmark; Vision Encoder: EVA-02-L (Transformer); Language Encoder: Vicuna v1.5 (LoRA); Action Decoder: Generative planner head
UAV-VLA (Sautenkov et al., 2025)	✔	✘	Zero-shot aerial mission VLA combining satellite and UAV imagery for scalable instruction-driven flight planning.	Dataset: Satellite + UAV flight logs; Vision Encoder: Molmo-7B-D (CLIP); Language Encoder: GPT-3; Action Decoder: Transformer-based path planner
Combat VLA (Chen et al., 2025a)	✔	✘	Ultra-fast tactical reasoning in 3D ARPG environments, achieving 50 times faster inference and human-level success.	Dataset: 3D ARPG combat logs; Vision Encoder: Qwen2.5-VL-3B; Language Encoder: Qwen2.5-VL-3B; Action Decoder: LLM-based planner head
HybridVLA (Liu et al., 2025)	✔	✘	Adaptive ensemble decoding that combines diffusion and autoregressive policies for robust multi-task generalization.	Dataset: RT-X trajectories + synthetic task fusion; Vision Encoder: CLIP ViT + DINOV2; Language Encoder: LLaMA-2; Action Decoder: Diffusion policy head
NORA (Hung et al., 2025)	✔	✘	Low-overhead VLA with integrated visual reasoning and FAST token decoding for real-time performance.	Dataset: OXE; Vision Encoder: Qwen-2.5-VL; Language Encoder: Qwen-2.5-VL; Action Decoder: FAST tokenizer head
SpatialVLA (Qu et al., 2025)	✔	✘	3D spatial encoding and adaptive action discretization to improve cross-robot manipulation generality.	Dataset: OXE; Vision Encoder: SigLIP; Language Encoder: PaliGemma (Gemma-2B); Action Decoder: Adaptive action grid head
MoLe-VLA (Zhang et al., 2025e)	✔	✘	Selective layer activation in a multi-stage ViT yields 5.6 time faster inference and +8% task success.	Dataset: RLBench + real-world trials; Vision Encoder: DINOv2, SigLIP; Language Encoder: LLaMA-2; Action Decoder: Diffusion head
JARVIS-VLA (Li et al., 2025b)	✔	✘	Open-world instruction following in 3D games via keyboard/mouse action prediction.	Dataset: Minecraft gameplay demos; Vision Encoder: ViT (in Llava-Next/Qwen2-VL); Language Encoder: Llava-Next/Qwen2-VL (transformer LLMs); Action Decoder: Key/mouse control head
UP-VLA (Zhang et al., 2025d)	✔	✘	Precise 3D spatial reasoning, achieving +33 % success on the CALVIN benchmark.	Dataset: CALVIN; Vision Encoder: CLIP-ViT; Language Encoder: Phi-1.5; Action Decoder: MLP policy head
Shake-VLA (Khan et al., 2025)	✔	✔	Modular bimanual VLA achieving 100% success on cluttered cocktail-mixing tasks.	Dataset: Cocktail mixing demos; Vision Encoder: YOLOv8, EasyOCR; Language Encoder: GPT-4o,Whisper-1; Action Decoder: Bimanual arm controller
MORE (Zhao et al., 2025a)	✘	✘	Scalable Mixture of Expert (MoE) enhanced RL framework for quadruped multi-task learning.	Dataset: Quadruped navigation/manipulation demos; Vision Encoder: CLIP-like; Language Encoder: Fuyu 8B ; Action Decoder: Mixture-of-Experts + LoRA adapter
DexGraspVLA (Zhong et al., 2025b)	✔	✘	Diffusion-based dexterous grasping with ≥ 90% zero-shot success across diverse objects.	Dataset: Self-collected Dexterous grasp data; Vision Encoder: DINOv2; Language Encoder: Qwen-VL, Qwen2.5-VL; Action Decoder: Diffusion policy head
DexVLA (Wen et al., 2025)	✔	✘	Cross-embodiment diffusion expert enabling rapid adaptation without per-task tuning.	Dataset: OXE, RLBench; Vision Encoder: Qwen2-VL (ViT), ResNet-50; Language Encoder:Qwen2-VL,DistilBERT; Action Decoder: Diffusion Transformer head
Humanoid-VLA (Ding et al., 2025)	✔	✘	Hierarchical VLA for full-body humanoid control, integrating perception and latent action planning.	Dataset: Self-collected humanoid robot episodes; Vision Encoder: Video Visual Encoder,Cross-Attention; Language Encoder: Llama3-70B; Action Decoder: Token-based Motion Decoder + RL Whole-Body Ctrlr
ObjectVLA (Zhu et al., 2025a)	✔	✘	End-to-end open-world object manipulation without task-specific data.	Dataset: RoboSpatial manipulation episodes; Vision Encoder: DinoX, DiVLA VLM; Language Encoder: DiVLA (LLM backbone); Action Decoder: Object-centric diffusion controller head
Gemini Robotics (Team, 2025b)	✔	✔	General-purpose VLA built on the Gemini 2.0 foundation, enabling long-horizon dexterous manipulation across diverse robot embodiments with zero-shot adaptability.	Dataset: Self-collected ALOHA2 demos + web-scale VL Dataset; Vision Encoder: Gemini 2.0 vision component; Language Encoder: Gemini 2.0 language component; Action Decoder: Local zero-shot policy head
ECoT (Zawalski et al., 2025)	✔	✔	Embodied chain-of-thought planning for interpretable, stepwise VLA control.	Dataset: Bridge v2 ; Vision Encoder: SigLIP, DINOv2; Language Encoder: LLaMA-2 7B; Action Decoder: Autoregressive VLA decoder with CoT module
OTTER (Huang et al., 2025a)	✔	✔	Zero-shot generalization via a frozen CLIP backbone and causal transformer action decoding.	Dataset: LIBERO; Vision Encoder: Frozen CLIP ViT; Language Encoder: CLIP text encoder; Action Decoder: Causal transformer delta-trajectory head
π-0.5 (Black et al., 2025)	✔	✔	Hierarchical VLA co-trained on real robot demos and web-scale vision-language data for robust household task generalization.	Dataset: self-collected 400h robot teleop + web VL dataset; Vision Encoder: SigLIP; Language Encoder: Gemma (2B/2.6B); Action Decoder: Flow Matching Head
OneTwoVLA (Lin et al., 2025)	✔	✔	Unified reasoning-acting framework that dynamically toggles between planning and control via decision tokens.	Dataset: Self-collected 16K reasoning-augmented robot episodes; Vision Encoder: same as pi-0 vla; Language Encoder:same as pi-0 vla; Action Decoder: Diffusion policy head
Helix (Team, 2025a)	✔	✔	First 200 Hz VLA for full humanoid control on embedded systems, enabling zero-shot task transfer.	Dataset: self-collected 200Hz teleop + sim logs; Vision Encoder: Pretrained VLM ; Language Encoder: Pretrained VLM ; Action Decoder: Fast transformer policy
Gemini Robotics On-Device (Parada and Team, 2025)	✔	✔	On-device optimized variant of Gemini VLA, delivering low-latency dual-arm and humanoid control on embedded hardware.	Dataset: Self-collected ALOHA2 + few-shot adaptation demos; Vision Encoder: Gemini SDK vision module; Language Encoder: Gemini SDK language module; Action Decoder: On-device optimized policy head
OE-VLA (Zhao et al., 2025c)	✔	✔	Curriculum-tuned LLaVA backbone with interleaved multimodal prompting for improved generalization across vision-language-action tasks.	Dataset: CALVIN; Vision Encoder: SigLIP-400M ViT; Language Encoder: Qwen-1.5 language module; Action Decoder: MLP token generator
SmolVLA (Shukor et al., 2025)	✔	✔	Ultra-lightweight VLA trained on community-contributed robot demonstrations, capable of real-time inference on CPU.	Dataset: 22.9K community episodes; Vision Encoder: SigLIP (VLM-2) visual backbone; Language Encoder: SmolVLM2 text backbone; Action Decoder: Chunked flow-matching head
EF-VLA (authors, 2025)	✔	✔	Early fusion of fine-grained CLIP visual tokens into the language-action pipeline, boosting zero-shot generalization.	Dataset: Self-collected real and simulated tasks; Vision Encoder: Frozen CLIP ViT; Language Encoder: Frozen CLIP text encoder; Action Decoder: causal transformer
PD-VLA (Song et al., 2025b)	✔	✘	First parallel decoding method with action chunking for VLA, achieving a 2.52 times speed-up without sacrificing control fidelity.	Dataset: Chunked trajectory demonstrations; Vision Encoder: CLIP-ViT-Large-Patch14-336 (LLaVA); Language Encoder: Vicuna-7B-v1.5 (LLaVA); Action Decoder: Fixed-point token predictor
LeVERB (Xue et al., 2025)	✔	✔	Dual-process latent VLA for whole-body humanoid control, achieving 58.5 % success on sim-to-real humanoid demos.	Dataset: sim-to-real humanoid demos; Vision Encoder: SigLIP ViT; Language Encoder: SigLIP text encoder; Action Decoder: Latent CVAE verb + transformer policy
TLA (Hao et al., 2025)	✔	✔	First language-grounded tactile-action model for high-precision contact tasks, with 85 % success on peg-in-hole task.	Dataset: TLA Data; Vision Encoder: ViT (Qwen2-VL); Language Encoder: Qwen2-VL; Action Decoder: Multimodal $\Delta x/\Delta y/\Delta z$ predictor
Interleave-VLA (Fan et al., 2025)	✔	✔	Model-agnostic wrapper enabling interleaved image-text instruction processing..	Dataset: Interleave-VLA data; Vision Encoder: Any base VLM (e.g., OpenVLA); Language Encoder: Any LLM (e.g., Pi-0); Action Decoder: Minimal interleaved-processing module
iRe-VLA (Guo et al., 2025)	✔	✔	Iterative RL and supervised fine-tuning pipeline for robust control and rapid generalization across embodiments.	Dataset: Franka-Kitchen, real Panda robot demos; Vision Encoder: BLIP-2 (pretrained VLM); Language Encoder: BLIP-2; Action Decoder: MLP action head after token learner

continue

Model Name	End-to-End	Comp. Focused	Main Contribution	Training Dataset and Model Components
TraceVLA (Zheng et al., 2025b)	✔	✔	Visual trace prompting to incorporate spatio-temporal cues, boosting task success by 3.5 time over OpenVLA.	Dataset: OXE + 150K trace-annotated demos; Vision Encoder: Phi-3-Vision with trace overlay; Language Encoder: Phi-3 LLM; Action Decoder: Quantized delta-motion tokens
OpenDrive VLA (Zhu et al., 2025b)	✔	✘	End-to-end driving VLA with semantic scene alignment and temporal abstraction for robust trajectory planning.	Dataset: Autonomous driving QA/planning benchmarks; Vision Encoder: ResNet-101 + Query Transformers; Language Encoder: Qwen2.5-Instruct (LLM); Action Decoder: Ego-vehicle action autoregressor
V-JEPA 2 (Assran et al., 2025)	✔	✘	Dual-stream self-supervised video JEPA enabling predictive planning in vision-language-action tasks.	Dataset: Droid video data; Vision Encoder: ViT (self-supervised) ; Language Encoder: LLM for QA/alignment; Action Decoder: Action-conditioned transformer predictor head
Knowledge Insulating VLA (Driess et al., 2025)	✔	✘	Implements insulation layers between vision-language and action modules, accelerating training and inference while maintaining generalization.	Dataset: Multi-domain VL datasets; Vision Encoder: PaliGemma (SigLIP); Language Encoder: PaliGemma (Gemma-2B) encoder; Action Decoder: Diffusion Modular policy head
GR00T N1 (Bjorck et al., 2025)	✔	✘	Self-collected Diffusion-based foundation model enabling unified humanoid control with policy tokenization.	Dataset: Multi-modal humanoid demonstrations; Vision Encoder: SigLIP-2 ViT (Eagle-2 VLM); Language Encoder: SmolLM2 (Eagle-2 VLM); Action Decoder: Generative diffusion transformer based planner
AgiBot World Colosseo (Bu et al., 2025)	✔	✘	Integrates multiple embodied datasets into a unified platform for scalable training and evaluation of VLA models.	Dataset: AgiBot World Data; Vision Encoder: PaliGemma (SigLIP); Language Encoder: PaliGemma (Gemma-2B); Action Decoder: Latent action planner + policy head
Hi Robot (Shi et al., 2025)	✔	✘	Hierarchical separation of planning and control for open-ended instruction following in complex environments.	Dataset: Self-collected Instruction-following data; Vision Encoder: PaliGemma-3B (SigLIP); Language Encoder: PaliGemma-3B (Gemma-2B); Action Decoder: Flow-Matching Action Expert
EnerVerse (Huang et al., 2025b)	✔	✘	World-model LLM for predictive future-space modeling, enabling long-horizon manipulation planning.	Dataset: self-collected Synthetic task fusion data; Vision Encoder:Pretrained VAE + Diffusion Generator; Language Encoder: Tokenized instruction prompt; Action Decoder: Diffusion Policy Head
FLaRe (Hu et al., 2024)	✔	✘	Large-scale RL fine-tuning framework generating robust, adaptive robot policies across domains.	Dataset: Multi-domain RL demonstrations; Vision Encoder: DinoV2; Language Encoder: Transformer policy (language tokens); Action Decoder: RL policy head
Beyond Sight (Jones et al., 2025)	✔	✘	Fuses heterogeneous sensor modalities via language-grounded attention to improve VLA generalization.	Dataset: self-collected Multi-sensor data; Vision Encoder: Multi-modal ViT; Language Encoder: Transformer (shared, task language input); Action Decoder: Transformer action head
GeoManip (Tang et al., 2025)	✔	✘	Encodes geometric constraints as model interfaces, enhancing robustness and precision in manipulation.	Dataset: Self-collected Simulated geometry tasks; Vision Encoder: VLM (GPT-4o) + Grounding-DINO; Language Encoder: GPT-4o; Action Decoder: Constraint solver head
Universal Actions (Zheng et al., 2025a)	✔	✘	Defines a universal action dictionary to standardize policy transfer and improve cross-task adaptability.	Dataset: Self-collected Cross-domain manipulation demos; Vision Encoder: Shared VLM (LLaVA-OneVion-0.5B); Language Encoder: LLaVA; Action Decoder: Unified action tokenizer head
RoboHorizon (Chen et al., 2025c)	✔	✘	LLM-enhanced multi-view environment modeling for robust long-horizon task planning.	Dataset: Self-collected Multi-view robot trajectories ; Vision Encoder: Multi-view transformer (ViT); Language Encoder: GPT-based planner; Action Decoder: DreamerV2 Actor-Critic RL Head
SAM2Act (Fang et al., 2025)	✔	✘	Utilizes SAM-based segmentation prompts with memory-augmented VLA for improved object-centric manipulation.	Dataset: SAM-labeled manipulation tasks; Vision Encoder: SAM2 segmentation encoder; Language Encoder: CLIP text encoder; Action Decoder: Memory-augmented policy head
LMM Planner Integration (Li et al., 2025d)	✔	✘	Merges LMM-based strategic planning with 3D skill policies for generalizable manipulation.	Dataset: skill library demos; Vision Encoder: DINO (2D semantics) + PointNext (3D); Language Encoder: CLIP Language Encoder; Action Decoder:3D Transformer head
VLA-Cache (Xu et al., 2025b)	✔	✘	Introduces token-caching to reuse computation across time steps, boosting inference efficiency.	Dataset: LIBERO; Vision Encoder: CLIP ViT; Language Encoder: LLaMA-2; Action Decoder: Cached inference head
Forethought VLA (Wu et al., 2025)	✔	✘	Aligns latent vision and action spaces for foresight-driven policy steering.	Dataset: Self-collected Latent alignment demonstrations; Vision Encoder: Phi-3 Vision; Language Encoder: LLama; Language; Action Decoder: Diffusion policy
GRAPE (Zhang et al., 2024c)	✔	✘	Preference-guided policy adaptation via personalized feedback alignment.	Dataset: Self-collected Preference-labeled demos; Vision Encoder: Dinov2; Language Encoder: LLaMA-2; Action Decoder: Autoregressive transformer head
HAMSTER (Li et al., 2025c)	✔	✘	Hierarchical skill decomposition to sequence multi-step manipulation actions.	Dataset: Self-collected Decomposed manipulation tasks; Vision Encoder: VILA-1.5-13B; Language Encoder: VILA-1.5-13B; Action Decoder: Robotic View Transformer Skill execution head
TempoRep VLA (Myers et al., 2025)	✔	✘	Use successor representation temporal encoding for compositional action planning.	Dataset: Self-collected Temporal demonstration sequences; Vision Encoder: ResNet-34 CNN; Language Encoder: retrained transformer (CLIP-style); Action Decoder: MLP (3x256) head on ResNet feature
ConRFT (Chen et al., 2025b)	✔	✘	Applies consistency regularized fine-tuning with reinforcement for stable policy learning.	Dataset: Self-collected data for fine-tuning; Vision Encoder: same as in octo; Language Encoder:same as in octo; Action Decoder: Reinforced policy head
RoboBERT (Wang et al., 2025)	✔	✘	Unified multimodal Transformer for end-to-end vision-language-action manipulation, pre-trained on diverse robot and language data.	Dataset: Self-collected Multi-domain robot demos; Vision Encoder: CLIP ViT; Language Encoder: BERT-base; Action Decoder: CNN-based Diffusion Policy Head
Diffusion Transformer Policy (Hou et al., 2024a)	✔	✘	Adapts diffusion-based transformer architectures to VLA policy learning, enabling robust multimodal action sampling.	Dataset: LIBERO + CALVIN; Vision Encoder: DINOv2; Language Encoder: CLIP Text Encoder; Action Decoder: Diffusion generator head
GEVRM (Zhang et al., 2025b)	✔	✘	Generative video modeling of goal-oriented tasks to enhance planning for visual manipulation.	Dataset: CALVIN; Vision Encoder: ResNet-34; Language Encoder: T5 Encoder; Action Decoder: Diffusion Policy
SoFar (Qi et al., 2025)	✔	✘	Introduces successor-feature orientation representations bridging spatial reasoning and robotic manipulation.	Dataset: Self-collected Orientation task demonstrations; Vision Encoder: Florence-2 (ViT-style), SAM; Language Encoder: CLIP Text Encode; Action Decoder: VLM (e.g., LLaVA or GPT-4o) for 6D goal pose, then motion planner
ARM4R (Niu et al., 2025)	✔	✘	Auto-regressive 4D transition model for predicting and planning manipulator trajectories.	Dataset: 76K videos from the Epic-Kitchens100 dataset ; Vision Encoder: ViT-Base; Language Encoder: CLIP text encoder; Action Decoder: 2-layer MLP
Magma (Yang et al., 2025)	✔	✘	Foundation multimodal agent model unifying vision, language, and action domains for end-to-end control.	Dataset: Self-collected Multimodal interaction dataset; Vision Encoder: ConvNeXt-XXlarge; Language Encoder: LLaMA-3-8B (decoder-only LLM); Action Decoder: Decoder-Only LLM Head (LLaMA-3-8B)
An Atomic Skill Library (Li et al., 2025a)	✔	✘	Constructs an atomic skill library for modular, data-efficient composition of robotic actions.	Dataset: Self-collected Skill primitive demonstrations; Vision Encoder: Prismatic VLM (scene description.), DINO-X (obj detection), SAM-2 (segmentation); Language Encoder: Prismatic, GPT-4 (for planning); Action Decoder: Skill executor module
VLAS (Zhao et al., 2025b)	✔	✘	Integrates speech-based LLM guidance for customizable voice-driven vision-language-action control.	Dataset: Speech-guided robot demos; Vision Encoder: CLIP ViT; Language Encoder: Vicuna (LLaMA-7B/13B); Action Decoder: Vicuna as an Autoregressive Action Decoder
ChatVLA (Zhou et al., 2025b)	✔	✔	Unified conversational VLA enabling natural-language and vision-driven interactive robot control with real-time multimodal feedback.	Dataset: Interactive human-robot demos; Vision Encoder: ViT + LoRA; Language Encoder: Qwen2-VL-2B (LLM); Action Decoder: mixture-of-expert action head, as in DiVLA)
RoboBrain (Ji et al., 2025)	✔	✘	Knowledge-grounded policy brain that maps abstract high-level plans to concrete multimodal actions across diverse tasks.	Dataset: Multi-domain robot and plan data; Vision Encoder: SigLIPr; Language Encoder: Qwen2.5-7B-Instruct (decoder-only LLM); Action Decoder: LoRA adapters for skill
SafeVLA (Zhang et al., 2025a)	✔	✘	Safety-aware VLA integrating constraint feedback through safe RL to ensure collision-free, reliable manipulation.	Dataset: Safety-scenario demonstrations; Vision Encoder: Modular (DINOv2, SigLIP, CLIP); Language Encoder: LLM (model-agnostic, e.g., T5, LLaMA, Qwen); Action Decoder: Safety-constraint policy head
CognitiveDrone (Lykov et al., 2025)	✔	✘	Embodied cognitive reasoning VLA for UAVs, combining vision-language understanding with autonomous flight planning.	Dataset: UAV mission logs; Vision Encoder: OpenVLA visual encoder (ViT-style) ; Language Encoder: OpenVLA's language encoder; Action Decoder: Transformer policy head
Diffusion-VLA (Wen et al., 2024)	✔	✔	Multimodal VLA framework unifying vision-language reasoning with diffusion-based policy for robust, generalizable manipulation across diverse robot embodiments.	Dataset: Multi-embodiment manipulation suites; Vision Encoder: SigLIP; Language Encoder: Qwen2-VL (2B/7B/72B); Action Decoder: Latent diffusion policy head + MLP

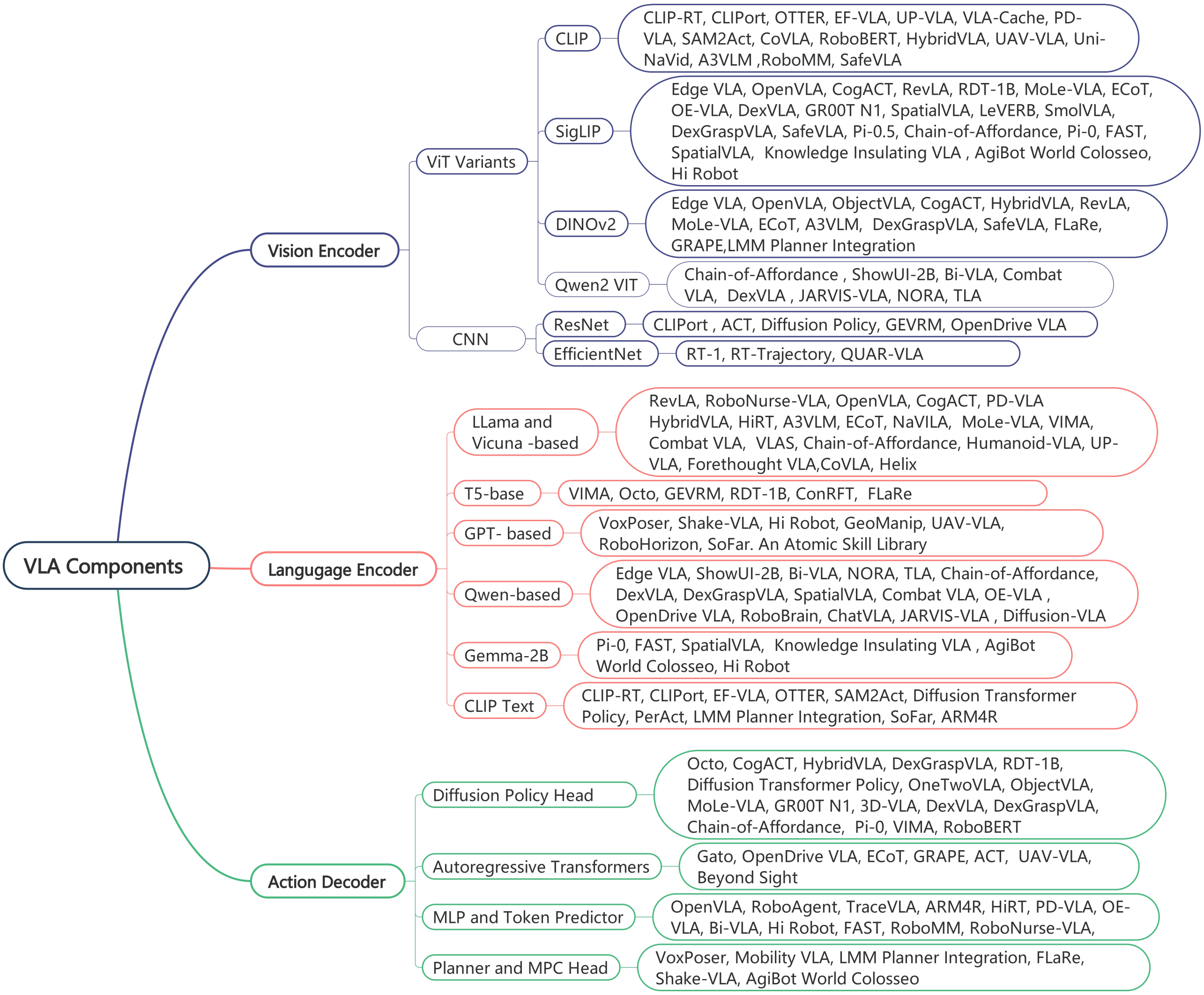

VLA Components
Vision Encoder
ViT Variants
CLIP
CLIP-RT, CLIPort, OTTER, EF-VLA, UP-VLA, VLA-Cache, PD-VLA, SAM2Act, CoVLA, RoboBERT, HybridVLA, UAV-VLA, Uni-NaVid, A3VLM ,RoboMM, SafeVLA
SigLIP
Edge VLA, OpenVLA, CogACT, RevLA, RDT-1B, MoLe-VLA, ECoT, OE-VLA, DexVLA, GR00T N1, SpatialVLA, LeVERB, SmolVLA, DexGraspVLA, SafeVLA, Pi-0.5, Chain-of-Affordance, Pi-0, FAST, SpatialVLA, Knowledge Insulating VLA , AgiBot World Colosseo, Hi Robot
DINOv2
Edge VLA, OpenVLA, ObjectVLA, CogACT, HybridVLA, RevLA, MoLe-VLA, ECoT, A3VLM, DexGraspVLA, SafeVLA, FLaRe, GRAPE,LMM Planner Integration
Qwen2 VIT
Chain-of-Affordance , ShowUI-2B, Bi-VLA, Combat VLA, DexVLA , JARVIS-VLA, NORA, TLA
CNN
ResNet
CLIPort , ACT, Diffusion Policy, GEVRM, OpenDrive VLA
EfficientNet
RT-1, RT-Trajectory, QUAR-VLA
Langugage Encoder
LLama and Vicuna -based
RevLA, RoboNurse-VLA, OpenVLA, CogACT, PD-VLA HybridVLA, HiRT, A3VLM, ECoT, NaVILA, MoLe-VLA, VIMA, Combat VLA, VLAS, Chain-of-Affordance, Humanoid-VLA, UP-VLA, Forethought VLA,CoVLA, Helix
T5-base
VIMA, Octo, GEVRM, RDT-1B, ConRFT, FLaRe
GPT- based
VoxPoser, Shake-VLA, Hi Robot, GeoManip, UAV-VLA, RoboHorizon, SoFar. An Atomic Skill Library
Qwen-based
Edge VLA, ShowUI-2B, Bi-VLA, NORA, TLA, Chain-of-Affordance, DexVLA, DexGraspVLA, SpatialVLA, Combat VLA, OE-VLA , OpenDrive VLA, RoboBrain, ChatVLA, JARVIS-VLA , Diffusion-VLA
Gemma-2B
Pi-0, FAST, SpatialVLA, Knowledge Insulating VLA , AgiBot World Colosseo, Hi Robot
CLIP Text
CLIP-RT, CLIPort, EF-VLA, OTTER, SAM2Act, Diffusion Transformer Policy, PerAct, LMM Planner Integration, SoFar, ARM4R
Action Decoder
Diffusion Policy Head
Octo, CogACT, HybridVLA, DexGraspVLA, RDT-1B, Diffusion Transformer Policy, OneTwoVLA, ObjectVLA, MoLe-VLA, GR00T N1, 3D-VLA, DexVLA, DexGraspVLA, Chain-of-Affordance, Pi-0, VIMA, RoboBERT
Autoregressive Transformers
Gato, OpenDrive VLA, ECoT, GRAPE, ACT, UAV-VLA, Beyond Sight
MLP and Token Predictor
OpenVLA, RoboAgent, TraceVLA, ARM4R, HiRT, PD-VLA, OE-VLA, Bi-VLA, Hi Robot, FAST, RoboMM, RoboNurse-VLA,
Planner and MPC Head
VoxPoser, Mobility VLA, LMM Planner Integration, FLaRe, Shake-VLA, AgiBot World Colosseo

Figure 8: This mind map presents the principal classes of vision encoders, language encoders, and action decoders employed in state-of-the-art VLA models. Only those encoder and decoder classes that are utilized by at least three different models are visualized, highlighting prevailing architectural trends across the VLA literature. The taxonomy categorizes representative models under each component family based on their dominant backbone; for example, ViT variants (such as; CLIP, SigLIP, DINOv2) and CNNs for vision encoding, LLaMA/Vicuna, T5-base, Qwen, and GPT-based models for language encoding, and diffusion or autoregressive transformers, MLP, and general Token predictors for action decoding. It should be noted that some models are listed under multiple encoder categories due to hybrid or fused architecture designs. For instance, models such as HybridVLA, OpenVLA, and DexGraspVLA appear under both SigLIP and DINOv2, as they integrate features from both backbones to enhance visual grounding and downstream task performance. This fusion-based design supports improved generalization, multi-view robustness, and more flexible multimodal alignment.

Models such as OpenVLA and Octo further incorporate proprioceptive tokens, while several systems (e.g., PerAct (Shridhar et al., 2022b), Helix (Team, 2025a)) support real-time feedback loops for continual correction. The rapid evolution and plug-and-play modularity of these architectures, where one can swap in a stronger ViT, a larger language model, or a more expressive diffusion sampler are driving to a new direction of instruction-driven autonomy for generalist robotic systems.

4.2. State-of-the-Art VLA Models

Table 1 is organized to provide a brief yet thorough overview of over a hundred VLA models related to robotic manipulation and instruction-driven autonomy. The first two columns detail the name and year of publication of each model. The next two columns *End-to-End* and *Component Focused* flag whether a model learns a direct mapping from raw visual and language input to control commands or instead concentrates on developing individual building blocks (for example, a better vision backbone or a more effective action sampler). The *Main Contributions* column then summarizes each work's core innovation, whether it introduced a novel fusion architecture, demonstrated a new training paradigm, or achieved state-of-the-art performance on a benchmark.

Each VLA model is based on four essential components: the training dataset, which provides foundational real-world task demonstrations or simulated episodes; the vision encoder, responsible for converting raw images or depth data into detailed feature maps; the language encoder, which maps instructions or annotations into a shared latent space; and the action decoder, which integrates these multimodal embeddings to produce the actual robot instructions, whether they are joint trajectories, discrete tokens, or overarching motion primitives. In Table 1, the final column specifies which dataset each model was trained on, which vision backbone it uses (e.g. CLIP-ViT, ResNet, EfficientNet), which text encoder it employs (e.g. T5, LLaMA, CLIP text), and what kind of action decoder it relies upon, e.g. Transformer head, diffusion policy, CVAE sampler (Xu et al., 2025a), making it straightforward to compare how different architectures assemble these building blocks.

4.3. Architectural Fusion Trends

Fig. 8 presents a comprehensive taxonomy of VLA model components, structured around three interconnected modules: vision encoders, language encoders, and action decoders. Within the vision encoder family, several prominent approaches exist. CLIP and SigLIP-based encoders are popular for their strong visual-text alignment through contrastive learning and are utilized in models such as CLIPort, RevLA, and Edge VLA. Other ViT variants such as DINOv2 and Qwen2 VIT, are used in models like Gato, Octo, HybridVLA, and Chain-of-Affordance for their ability to model long-range spatial dependencies and high-level visual semantics. CNN-based encoders such as ResNet and EfficientNet appear in models like CLIPort, ACT, RT-1, and QUAR-VLA.

Language encoders show similar architectural diversity. LLaMA and Vicuna-based language encoders are widely used in models such as RevLA, OpenVLA, and HybridVLA for instruction understanding and zero-shot reasoning. T5-style models appear in VIMA, Octo, and FLARE, offering flexible encoder-decoder structures for sequence generation. GPT-based and Qwen-based encoders, such as those used in VoxPoser, Edge VLA, and DexVLA, balance generalization and compact deployment. Gemma-2B language encoders are found in Pi-0 and FAST, while specialized solutions like CLIP Text encoders are used in CLIPort and PerAct for minimal alignment tasks.

In action decoders, diffusion-based transformers are a leading choice for models like Octo, HybridVLA, and DexGraspVLA, as they offer fine-grained, temporally smooth control via iterative denoising. Autoregressive Transformer heads, such as those in Gato, OpenDrive VLA, and GRAPE, generate action sequences step-by-step, optimizing real-time responsiveness. Several models including VoxPoser, LMM Planner Integration, and FLARE embed Model Predictive Control or specialized planning heads to support decision-making in dynamic tasks. MLP or token predictor heads are used in OpenVLA, TraceVLA, and RoboMamba for efficient low-level control.

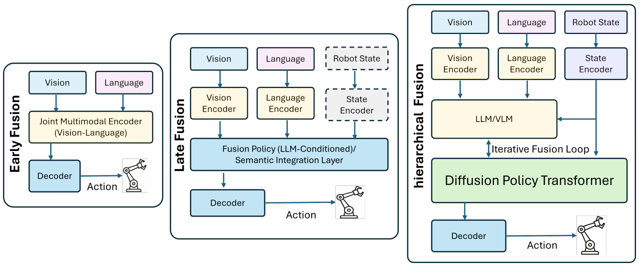


Figure 9: Fusion paradigms in VLA architectures. (a) *Early Fusion:* Visual and linguistic inputs are combined at the input stage through a *Joint Multimodal Encoder* to form a shared representation. (b) *Late Fusion:* Separate vision, language, and state encoders are merged by a *Fusion Policy (LLM-conditioned)* or semantic integration layer before decoding. (c) *Hierarchical Fusion:* Multimodal interaction occurs across perceptual, semantic, and control levels via bidirectional links between encoders, an LLM/VLM reasoning core, and a *Diffusion Policy Transformer*. Hierarchical designs enable iterative refinement, stronger grounding, and improved robustness in embodied control.

Our evaluation of VLA architectures identifies three dominant types of fusion frameworks, *early*, *late*, and *hierarchical* as illustrated in Fig. 9.

(a) Early Fusion: Visual and linguistic inputs are integrated at the input stage through a shared multimodal encoder that jointly processes tokens from both modalities. Models such as *CLIPort* and *EF-VLA* exemplify this approach, fusing text and image embeddings early to achieve strong perception instruction alignment and compact latent representations.

(b) Late Fusion: Vision, language, and robot-state signals are encoded independently and later merged by a high-level semantic integration layer or an LLM/VLM that conditions policy generation. This modular design, employed in architectures such as *RT-2* and *OpenVLA*, supports scalable pretraining and cross-task generalization but offers weaker fine-grained grounding between modalities compared to deeper fusion hierarchies.

(c) Hierarchical Fusion: Fusion is distributed across multiple layers of abstraction, enabling iterative interaction between perception, language understanding, and control. Architectures such as *GR00T-N1*, *DexVLA*, and *HybridVLA* implement this strategy by coupling transformer-based reasoning modules with diffusion- or flow-based policy decoders, allowing continuous semantic control feedback. This approach enhances contextual reasoning, improves real-to-sim transfer, and increases robustness across diverse embodied robotic tasks.

The qualitative synthesis above encourages a more detailed examination of the practical impact of these design choices.

4.4. Quantitative Meta-Analysis of Architecture Performance Relationships

Robotic manipulation presents specific challenges that require the coordinated integration of visual perception, natural language grounding, and closed-loop action execution. VLA model architectures are therefore expected to perceive objects in cluttered environments, interpret task semantics (e.g., open drawer → grasp → insert), and maintain stable interaction under physical disturbances. These requirements suggest that architectural design choices, including multimodal fusion hierarchy, encoder scale, and decoder dynamics, are directly tied to manipulation success. To investigate these relationships systematically, we perform a quantitative meta-analysis of benchmark observations from reviewed models collected from recent literature. The Details of the key concepts and of the below computation details can be found in the Appendix A and B.

For each model, we extract the main success metric reported in its original source, normalize it to the interval $[0, 1]$, and apply a difficulty adjustment so that models evaluated on more challenging settings are not unfairly penalized. This correction is derived from a Difficulty Index that incorporates task complexity (C_{task}), modality richness (C_{mod}), and dataset scale ($\log N$). The resulting score, denoted `Normalized success`, gives greater weight to performance on complex, multimodal, long-horizon evaluations and assigns lower importance to performance on simple or single-modal tests, ensuring comparability across heterogeneous benchmarks. This corrected and clipped value serves as the dependent variable Y in the regression analysis.

The regression model relates normalized performance to architectural and dataset characteristics as follows:

$$\begin{aligned} Y = {} & \beta_0 + \beta_1 D_f + \beta_2 S_v + \beta_3 S_\ell + \beta_4 C_{\text{task}} + \beta_5 C_{\text{mod}} \\ & + \beta_6 \mathbb{1}_{\text{diffusion}} + \beta_7 \mathbb{1}_{\text{flow}} + \beta_8 \mathbb{1}_{\text{hierarchical}} \\ & + u_{\text{bench}} + \varepsilon. \end{aligned} \tag{4}$$

where D_f measures fusion depth design (early, late, or hierarchical); S_v and S_ℓ capture the scale of visual and linguistic encoders, respectively; C_{task} reflects contact and sequential difficulty; and C_{mod} denotes modality richness. The indicator terms $\mathbb{1}_{\text{diffusion}}$, $\mathbb{1}_{\text{flow}}$, and $\mathbb{1}_{\text{hierarchical}}$ represent binary variables identifying diffusion-based decoders, flow-matching decoders, and explicitly hierarchical fusion architectures, respectively, with symbolic or MLP-based controllers serving as the baseline. The random term u_{bench} accounts for *benchmark-specific heterogeneity* across datasets and evaluation suites, and ε denotes the residual noise. All continuous predictors are standardized to allow direct comparison of effect magnitudes. Importantly, the regression does not generate success scores; the values of Y are sourced directly from paper-reported results after applying the above stated difficulty index-based rubric. The regression simply learns how architectural choices statistically associate with these normalized outcomes. The fitted coefficients and their 95% confidence intervals populate the forest plot.

Fig. 10 depicts standardized regression coefficients grouped by functional category as: *Decoder Policy* ($\mathbb{1}_{\text{diffusion}}$, $\mathbb{1}_{\text{flow}}$), *Task Complexity* (C_{task}, C_{mod}), *Model Scale* (S_v, S_ℓ), and *Architecture Design* (D_f, $\mathbb{1}_{\text{hier}}$). A vertical reference line at $\beta = 0$ facilitates assessment of statistical significance. The analysis shows that diffusion-based action decoders produce the largest positive contribution to success, suggesting that smooth, feedback-aware sampling significantly improves closed-loop correction during grasping and alignment tasks. Flow-matching decoders show a comparable, though slightly lesser, advantage. Hierarchical fusion strategies demonstrate a strong positive influence, indicating that maintaining semantic grounding throughout the control pipeline enhances error recovery and long-horizon stability. In contrast, symbolic and MLP-style controllers correlate negatively with success, reflecting their limited robustness in the presence of real-world uncertainty. The scale of vision and language encoders contributes positively but modestly, implying that enlarging perceptual or linguistic capacity alone does not compensate for weak fusion or action models. Overall, the forest plot indicates that decoder dynamics and hierarchical fusion are the primary architectural drivers of manipulation success, while encoder scale contributes marginally and task-level complexity plays little role after normalization.

To examine broader architectural trends beyond linear effects, we analyze continuous relationships between model scale, fusion depth, and predicted performance. As illustrated in Fig. 11, deeper fusion consistently leads to higher manipulation success, suggesting that iterative cross-modal interaction between perception and task semantics is critical for informed action selection. Larger visual encoders offer improved robustness to occlusions and cluttered scenes, although these gains are less pronounced than those obtained through deeper fusion. In general, the results indicate that architectural decisions governing where and how modalities interact matter more than simply increasing model size.

We further study the latent structure underlying the observed performance variance by applying factor analysis to the model features (Fig. 12). Three dominant performance axes emerge. Factor 1 (*Architecture*) is characterized by strong loadings on fusion depth and encoder scale, defining the principal direction of variance. This factor reflects the ability of a system to propagate semantic and geometric information through the action hierarchy. Factor 2 (*Scale*) captures dataset size and multimodality, aligning with generalization benefits obtained from broader experience. Factor 3 (*Performance*) accounts for stability differences under contact-rich conditions where minor deviations must be corrected continuously.

Taken together, these analyses demonstrate that reliable manipulation performance is primarily influenced by multimodal fusion strategies and action decoder dynamics rather than parameter scaling alone. The results support a mechanistic view in which VLA success depends on the quality of perception-action grounding: architectures that repeatedly integrate state, vision, and task semantics throughout the policy pipeline achieve superior error recovery and contact stability. Thus, deeper cross-modal interaction serves as the primary driver of real-world manipulation capability,

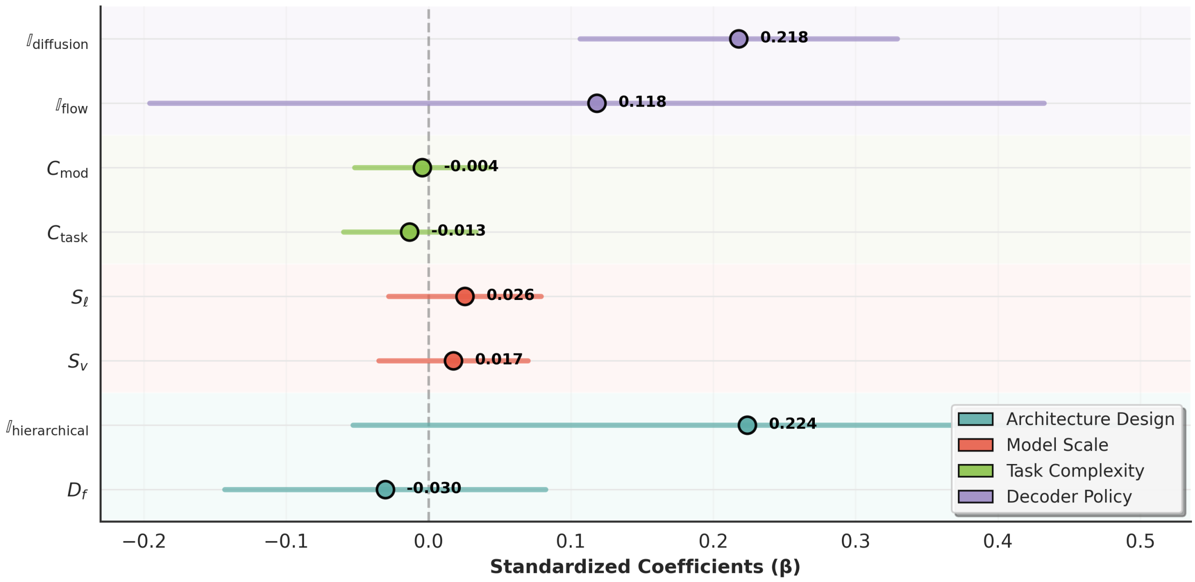


Figure 10: Architecture-to-performance regression analysis. The forest plot displays standardized regression coefficients showing that diffusion-based decoders and hierarchical fusion exert the strongest positive associations with normalized manipulation success, while shallow fusion depth and symbolic or MLP-style controllers correlate negatively with robustness across contact-rich tasks.

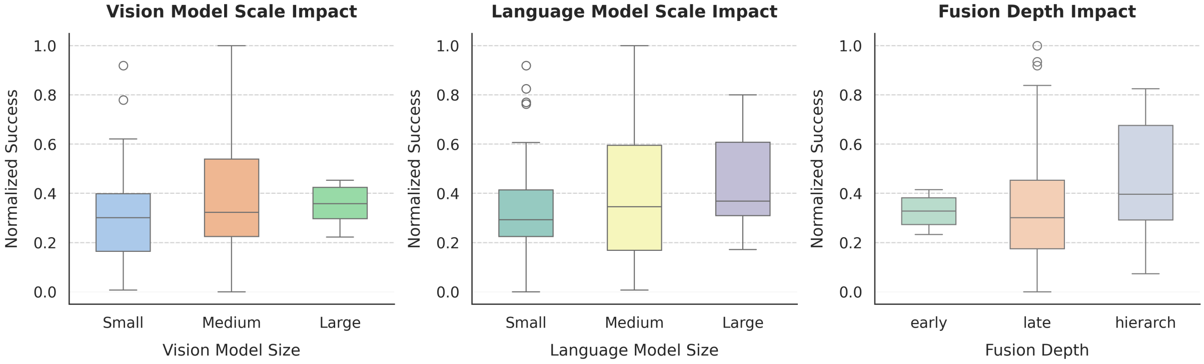


Figure 11: Impact of model scale and fusion depth on success rate. Deeper fusion and hierarchical architectures consistently outperform shallow or component-based systems, whereas pure scaling of encoder capacity provides comparatively incremental improvements.

while model and dataset scale act as secondary enablers that enhance generalization without fundamentally altering policy reliability.

4.5. Unified Theoretical Framework and Empirical Validation

To connect architectural trends with their functional role in robotic behavior, we develop an information-theoretic framework describing how deeper fusion hierarchies enhance decision confidence and task stability in VLA systems. As shown in Fig. 13, progressively structured multimodal fusion reduces uncertainty in action generation and strengthens the coupling between perception, language, and control, producing more coherent and reliable robot behavior.

To explain why deeper and more structured fusion architectures improve the empirical results reported in Sec. 4.4, we introduce an information-theoretic formulation that quantifies how multimodal representations progressively reduce uncertainty in robotic action generation. Let V, L, and S denote the visual, linguistic, and proprioceptive modalities, $\{Z_k\}_{k=1}^{K}$ the latent fusion layers, and A the executed actions. Each fusion layer integrates sensory and semantic cues to form an updated belief about the appropriate control action. The contribution of each fusion stage is expressed as the entropy reduction over the action distribution:

$$\begin{aligned}\Delta H_k &= H(A \mid Z_{k-1}) - H(A \mid Z_k),\\ \sum_{k=1}^{K} \Delta H_k &= H(A) - H(A \mid Z_K).\end{aligned} \tag{5}$$

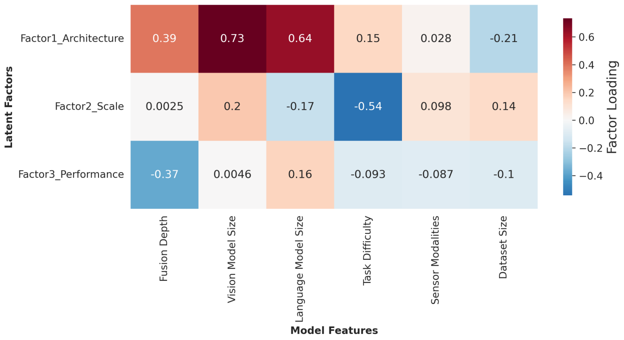


Figure 12: Latent factor structure underlying VLA model performance. Factor 1 (*Architecture*) exhibits the strongest loadings on vision and language model scale as well as fusion depth, indicating that architectural complexity primarily drives cross-model variance. Factor 2 (*Scale*) reflects dataset and modality diversity, while Factor 3 (*Performance*) captures residual variance associated with fusion efficiency and task difficulty. Together, these latent dimensions reveal how structural capacity, data richness, and task complexity jointly shape VLA success.

where $H(\cdot)$ denotes Shannon entropy of the robot's action policy. ΔH_k measures how much uncertainty is removed when moving from fusion layer Z_{k-1} to Z_k. A larger cumulative $\sum_k \Delta H_k$ indicates that the robot's internal representation increasingly constrains the action space, enabling more confident control decisions, particularly in long-horizon, contact-rich manipulation where uncertainty must be minimized.

We further define an efficiency measure that normalizes information gain by the computational resources required at each layer:

$$\eta_k = \frac{I(Z_k; V, L, S) - I(Z_{k-1}; V, L, S)}{\mathrm{FLOPs}_k}, \tag{6}$$

where $I(\cdot\,;\cdot)$ represents mutual information between the fused latent state and the input modalities, and FLOPs_k is the computational cost at fusion layer k. This ratio reflects how effectively each layer converts perception-language correlation into useful control information per unit cost. Higher η_k values, especially in hierarchical fusion, imply more efficient cross-modal binding: the robot learns to align semantic instructions with geometric affordances and proprioceptive feedback without redundant computation.

Finally, we define a fusion-energy measure that quantifies how much multimodal coupling improves policy likelihood compared to unimodal baselines:

$$E_{\text{fusion}} = \mathbb{E}\Big[-\log p_{\text{full}}(A \mid V, L, S) + \log p_{\text{ablated}}(A \mid V, L, S)\Big], \tag{7}$$

where p_{full} and p_{ablated} denote the action likelihoods under full and reduced-modality models, respectively. A higher E_{fusion} implies stronger information integration, resulting in more stable policy updates and fewer cascading control failures during execution. Empirically, diffusion and flow-based decoders maximize this functional, confirming that deeper hierarchical fusion architectures increase both representational richness and task-level reliability. The details about how these quantaties are computed is explained in Appendix C

These quantities, ΔH_k, η_k, and E_{fusion}, form a unified theoretical perspective linking architecture design to manipulation success: robots act more reliably when visual perception, language constraints, and contact feedback are fused repeatedly throughout the control hierarchy. They explain why hierarchical architectures work by repeatedly reducing uncertainty, maximizing cross-modal efficiency, and optimizing energy integration, producing more coherent and robust robotic actions in complex manipulation environments.

Fig. 13 visualizes these theoretical quantities across diverse VLA architectures. Fig. 13-(a) shows that hierarchical fusion achieves the greatest entropy reduction (ΔH_k), validating that deeper multimodal coupling progressively constrains the action distribution and improves execution stability. Fig. 13-(b) depicts cross-modal attention efficiency (η_k), where hierarchical models generate higher and more consistent efficiency, aligning semantic intent with geometric and proprioceptive cues more effectively and at lower computational cost. Fig. 13-(c) demonstrates that the fusion-energy functional (E_{fusion}) correlates with normalized task success, confirming that increased multimodal coupling enhances policy likelihood and trajectory stability. Together, these results empirically substantiate the proposed information-theoretical formulation: deeper hierarchical fusion reduces uncertainty, enhances representational efficiency, and improves energy performance trade-offs, leading to more robust and reliable robotic actions in real-world manipulation tasks.

5. Datasets for Multi-Modal Fusion

The foundation of VLA models lies in high quality, diverse training datasets. These datasets are crucial as they expose models in the complete range of real and simulated environments, ensuring a tight alignment of visual elements, natural language instructions, and control (James et al., 2020). These datasets allow VLAs to learn complex cross-modal correlations, such as how language complexities (e.g., 'gently place') affect motion smoothness without relying on manually prepared heuristics. We first introduce the unified dataset schema that underlies the VLA training pipelines, then survey the most influential public datasets, and finally apply a comprehensive benchmarking strategy to assess the scale, modality coverage, and complexity of each dataset.

5.1. Dataset Format

Structured overview of the general dataset format commonly used in VLA training pipelines. is illustrated in Fig. 14. It highlights the systematic organization of multimodal data into three primary streams: visual, language, and action/control, which collectively facilitate the training and evaluation of VLA models.

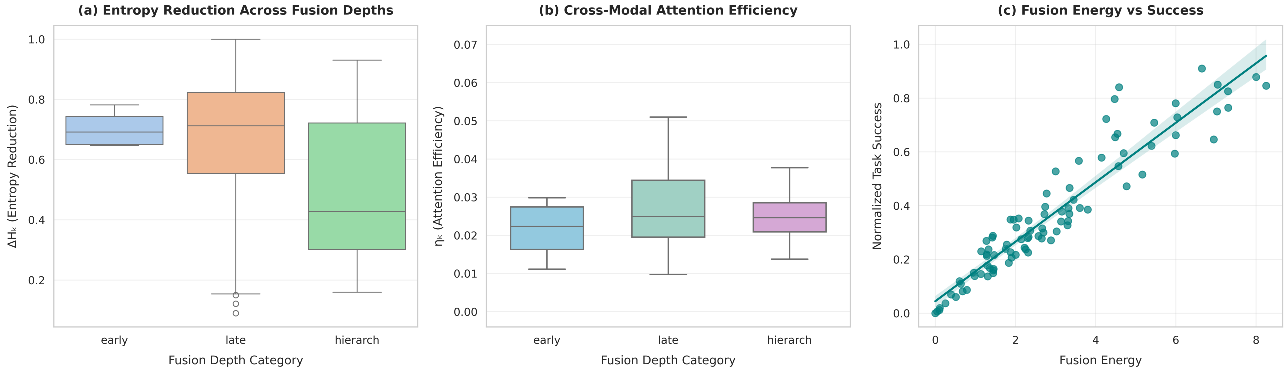


Figure 13: Quantitative visualization of fusion dynamics across VLA model architectures. (a) Entropy reduction (ΔH_k) increases from early to late and reaches its highest values in hierarchical fusion, indicating progressive uncertainty reduction in action selection. (b) Cross-modal attention efficiency (η_k) measures information gain per computational cost, showing that hierarchical fusion achieves higher and more consistent efficiency compared with early or late fusion. (c) Fusion energy (E_{fusion}) correlates strongly with normalized task success, confirming that architectures with greater multimodal coupling yield higher policy likelihood and more stable robotic performance.

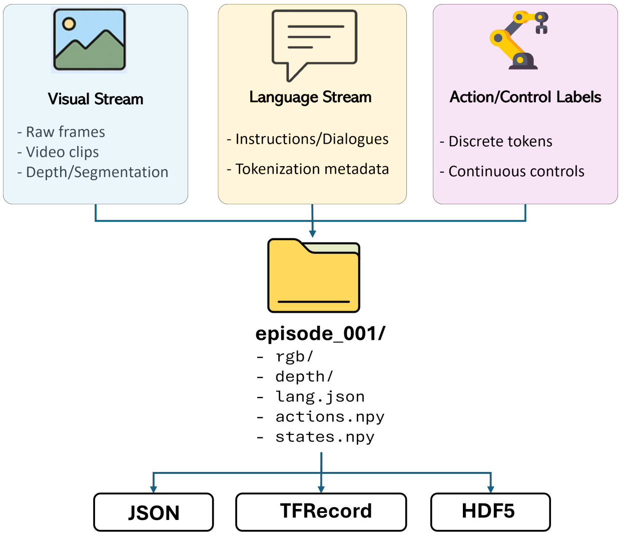


Figure 14: Schematic of the unified VLA training data format. Visual observations, language instructions, and action/control signals are grouped into episode directories and serialized into standardized storage formats (JSON, TFRecord, and HDF5), enabling efficient and scalable data loading for end-to-end model training.

The *Visual Streams* comprise raw RGB frames, video snippets, and optionally, depth maps and segmentation masks. These visual inputs provide essential spatial and contextual data to perception modules in VLA architectures. Typically, the data in this stream is stored in standard image or video formats like JPEG or PNG for individual frames and MP4 or similar formats for video snippets.

The *Language Streams* incorporate natural-language instructions or dialogues alongside tokenization metadata (such as token offset). These textual annotations are essential for instructing and conditioning robotic actions and are commonly stored in lightweight, structured formats such as JSON or plain text files. The presence of tokenization metadata facilitates efficient textual processing, enabling direct integration with transformer-based language models.

The *Action/Control Labels* include discrete action tokens (e.g. categorical commands like "move forward" or "turn left") and continuous control vectors representing joint positions or end-effector trajectories. These action labels provide explicit supervision signals for model output and are typically stored as NumPy arrays or encoded within structured data containers.

All three modality streams are systematically integrated into standardized episode-level directories (e.g., `episode/`), where visual data reside in subdirectories such as `rgb/` and `depth/`, accompanied by `lang.json`, `actions.npy`, and `states.npy`. Each episode folder can then be serialized in formats like: *JSON* for lightweight, human-readable metadata; *TFRecord/TF-Example* for high-throughput, sharded training; or *HDF5* for efficient random access to synchronized frames, actions, and state arrays, enabling balance readability, I/O performance, and scalability in their training pipelines.

5.2. Major VLA Datasets

Table 2 summarizes the progress of VLA datasets, highlighting how each dataset advances autonomy by varying in scale, modality, and task complexity. Early collections such as EmbodiedQA and R2R focus on discrete decision making in constrained environments, offering simple state-action mappings suitable for evaluating baseline policy architectures (e.g., PACMAN, Speaker-Follower). As we move into 2020-2022, datasets like ALFRED, RLBench, and CALVIN introduce longer-horizon tasks and richer sensory streams combining RGB, depth, proprioception, and natural language instructions to stress test hierarchical planning and subgoal decomposition methods (e.g., C2F-ARM, VIMA, RT-2). These mid-generation datasets bridge the gap between symbolic planners and end-to-end learning, enabling comparative analyses of model-based control versus learned policies under simulated dynamics.

Table 2: Overview of main VLA datasets used in robotic manipulation and embodied AI research. For each dataset, we list the release year, dataset size, distinctive characteristics, and the data storage format.

Dataset	Size	Distinctive Characteristics	Data Format
EmbodiedQA (Das et al., 2018)	5,000+ QA episodes across 750+ 3D scenes	Goal-directed visual question answering in House3D with object and room diversity	JSON-formatted question-answer pairs, egocentric RGB frame sequences, and agent trajectories
R2R (Anderson et al., 2018)	7,189 unique paths with 21,567 natural language instructions	Real-world vision-language navigation using Matterport3D with path diversity and crowd-sourced instructions	JSON files with instructions and navigation paths; panoramic JPEG frames and viewpoint graph metadata
ALFRED (Shridhar et al., 2020)	8,055 expert demonstrations with 25,743 language directives	Language-conditioned household manipulation tasks in AI2-THOR 2.0 across 120 indoor scenes	Per-demonstration folders with egocentric RGB frames, ground-truth interaction masks, and language annotations; metadata and action/state sequences in JSON format
RLBench (James et al., 2020)	Expert demonstrations available for 100 vision-based manipulation tasks	Large-scale few-shot and imitation learning benchmark in PyRep simulation	Pickled demos include `joint_positions`, `camera_images`, `task_description`, and proprioceptive states.
CVDN (NDH) (Thomason et al., 2019)	7,415 navigation-from-dialog-history instances from 2,050 dialogs	Vision-and-Dialog Navigation benchmark requiring agents to act based on dialog history	JSON annotations with dialog turns, navigation paths, image features, and speaker roles
TEACh (Padmakumar et al., 2021)	3,047 successful two-agent gameplay sessions	Multiturn dialog-driven household task completion in AI2-THOR	JSON transcripts aligned to visual frames, with egocentric RGB, object masks, action logs, and benchmark CSV splits
DialFRED (Gao et al., 2022a)	53,000+ human-annotated QA pairs across 34,000+ tasks	Dialogue-enabled embodied instruction following on augmented ALFRED subgoals	Per-task `dialogue.json` with human and oracle QAs, action traces, subgoal templates, and frame alignment
Ego4D (Grauman et al., 2022)	3,670 h of first-person video	Large-scale, real-world egocentric dataset with diverse scenarios and modalities	MP4 video clips; JSON-based narrations and annotations; HDF5/LMDB indices; multilingual narration files.
CALVIN (Mees et al., 2022)	5,000+ demonstrations	Long-horizon, language-conditioned robotic manipulation tasks	HDF5 archives with synchronized RGB-D frames, proprioception, action sequences, and natural language instructions.
DROID (Khazatsky, 2024)	76k demonstrations; 564 scenes, 86 tasks	High-diversity language-conditioned robot manipulation	RLDS-formatted RGB-D data, stereo video, camera calibrations, language annotations, and robot state/action logs.
Open X-Embodiment (Collaboration et al., 2025)	1M+ trajectories, 500+ skills, 22 robot types	Large-scale, multi-embodiment, multi-skill manipulation dataset	Sharded TFRecords with RGB/depth frames, language instructions, action vectors; YAML metadata and RLDS format.
RoboSpatial (Song et al., 2025a)	1M images, 5K 3D scans, 3M spatial QA pairs	2D-3D paired spatial reasoning dataset	RGB images, 3D scans, and relational graph annotations in support of spatial understanding benchmarks.
CoVLA (Arai et al., 2024)	83.3 h real-world driving video, 6M frames	Time-aligned vision-language-action dataset	Synchronized RGB video, GPS/IMU-based trajectories, and auto-generated captions using rule-based and VLM methods.
TLA (Hao et al., 2025)	30K contact-rich peg-in-hole demonstrations	Tactile-language-action alignment for precise insertion and assembly	ROS bag files with synchronized `camera/`, `tactile/`, `lang.json`, and `trajectory.csv` recordings.
BridgeData V2 (Walke et al., 2023)	60,096 trajectories (50,365 teleoperated; 9,731 scripted)	Multi-skill goal- and language-conditioned manipulation dataset	TFRecords with RGB images, natural language instructions, and continuous 7-DoF action vectors; includes both human and scripted demonstrations.
LIBERO (Liu et al., 2023)	130 tasks: 10 spatial, 10 object, 10 goal, 100 lifelong	Lifelong VLA benchmark for procedural and declarative knowledge transfer	JSON and Parquet files with RGB images, language instructions, action trajectories, and structured metadata.
Kaiwu (Jiang et al., 2025)	1M multimodal robotic episodes	Real-world, multi-embodiment dataset for dexterous manipulation with natural language commands	Per-episode HDF5 files with synchronized RGB, depth, 3D skeletons, tactile, EMG, gaze, IMU, audio, language, and motion capture data.
PLAICraft (He et al., 2025)	10,000 + hours of multiplayer Minecraft gameplay across 5 modalities	Open-ended multiplayer interaction with emergent task structures and voice-aligned social play	JSON-encoded multimodal streams (RGB, audio, keyboard, mouse) with millisecond alignment
AgiBot World (Bu et al., 2025)	1M+ multimodal dual-arm trajectories	Open-source platform for long-horizon generalist policy learning	ROS-based: RGB-D, fisheye, tactile, proprioception, language, error annotations, and dexterous control logs.
Robo360 (Liang et al., 2023)	2K+ real trajectories, 86 calibrated views, 100+ diverse objects	Multimodal dataset for dynamic NeRF, imitation learning, and control	Synchronized RGB videos, depth maps, audio, robot proprioception, and control signals per frame.
REASSEMBLE (Sliwowski et al., 2025)	4,551 contact-rich demonstrations, 17 objects, 781 minutes	Multimodal (RGB, audio, event, force/-torque, proprioception)	Synchronized multistream recordings from RGB cameras, event camera, microphones, force/torque sensors, and proprioceptive signals, collected during haptically teleoperated assembly and disassembly tasks based on NIST benchmark boards.
RoboCerebra (Han et al., 2025)	100K long-horizon trajectories across 1K+ tasks	System-2-level reasoning and generalization in real-world-scale settings	Structured plan logs, visual transitions, failure annotations, and dense subtask labels from human-verified demonstrations and multi-stage task generation.
IRef-VLA (Zhang et al., 2025c)	11.5K rooms, 7.6M spatial relations, 4.7M instructions	Imperfect referential grounding in 3D indoor scenes	Per-room scene graphs, free-space maps, and affordance annotations with synthetic and imperfect language queries.
Interleave-VLA (Fan et al., 2025)	210K episodes (13M frames)	Interleaved vision-language instruction execution	Mixed-format episodes with images, sketch overlays, and text prompts aligned with action sequences.
RoboMM (Yan et al., 2024)	30K simulated episodes + 5K real-world trials	Multimodal fusion of vision, language, proprioception, and touch	HDF5 per episode with `rgb/`, `depth/`, `tactile.csv`, `instructions.json`, and `action_sequence.json`.
ARIO (Wang et al., 2024b)	50K simulated episodes + 5K real-world trials	Contact-rich manipulation with tactile, audio, and proprioceptive feedback	Per-episode HDF5 archives with `rgb/`, `depth/`, `tactile.csv`, `instructions.json`, and `action_sequence.json`.

From 2023 onward, the field shifts to truly multimodal control challenges. Datasets such as DROID and Open X-Embodiment embed synchronized RGBD, language, and multi-skill trajectories, facilitating evaluation of sensor fusion strategies and real-time feedback controllers. Large-scale egocentric corpora like Ego4D and CoVLA offer real-world visual streams that drive research on robust perception-action loops under unpredictable dynamics. Recent contact-rich datasets such as ARIO, TLA, RoboMM, and REASSEMBLE integrate high-frequency haptic and force/torque feedback alongside vision and language, enabling fine-grained impedance control and hybrid model-predictive schemes for deformable-object manipulation. Highly multimodal and large-scale datasets such as Kaiwu, PLAICraft, AgiBot World, and Robo360 support open-ended, long-horizon, and real-world tasks with diverse sensor suites including tactile, audio, proprioceptive, and multi-view data. By standardizing annotation formats (HDF5 bundles, ROS bags, TFRecords) and pairing each collection with representative baselines (e.g., SayCan, HapticBERT, MM-FusionNet), Table 2 provides a detailed overview of these datasets across a continuum of task complexity, modality richness, and real-world fidelity.

5.3. Benchamrk VLA Datasets

In order to benchmark, we map each major VLA dataset onto a two-dimensional plane spanned by task complexity and modality richness, illustrated in Fig. 15. The x-axis captures how challenging each dataset's manipulation tasks are, ranging from simple single-step actions to long-horizon, multiskill sequences. The y-axis shows the modality richness, from minimal (dual modalities: text and image) to comprehensive (up to seven modalities including audio, video, robot proprioception, control, depth, haptics, and language).

To quantify these dimensions systematically, we assign scalar scores to each dataset reflecting their task complexity and modality richness. Task complexity, denoted as C_{task}, incorporates:

- Average number of low-level actions per episode (T). This captures how many primitive control commands are grouped together in a typical task (e.g., grasp, lift, move).
- Number of distinct high-level skills (S). This enumerates different semantic subtasks (e.g., open drawer, pick object).
- Degree of sequential task dependency (D). This denotes the fraction of tasks that require strict ordering of subtasks; $D \in [0, 1]$.
- Linguistic abstraction level (L). Quantifies the average linguistic complexity (e.g., vocabulary size or syntactic depth) of the instruction set; $L \in \mathbb{R}^{+}$.

These attributes are integrated via the following weighted model:

$$C_{\text{task}}(\mathcal{D}) = \alpha_1 \log(1 + T) + \alpha_2 S + \alpha_3 D + \alpha_4 L, \qquad (8)$$

where $\alpha_i > 0$ for $i = 1, \dots, 4$ are weights that normalize each term to commensurate scales and can be tuned to reflect the emphasis on action length, skill diversity, sequential structure, or language complexity. For our benchmark, we set all weights equal to one.

Modality richness, captured by the score C_{mod}, integrates four factors reflecting the scope and quality of sensory input:

- Number of distinct modalities (M), Such as vision, depth, haptics, and language.
- Mean quality $Q = \frac{1}{M} \sum_{i=1}^{M} Q_{m_i}$, Where Q_{m_i} for $i = 1, \dots, M$ are the modality-specific quality scores. Q_{m_i} can be determined by expert annotation, automated signal-to-noise ratio analysis, or based on dataset documentation and previous benchmark studies. For this work, we use a mixture of empirical review and published specifications to assign scores in the range $[0.6, 0.95]$, reflecting the typical range of public datasets.
- Fidelity of temporal alignment across modalities (A), Measures how tightly modalities are synchronized (e.g., frame-accurate vision-language pairing), with $A \in [0, 1]$.
- Presence of reasoning-critical modalities (R): Such as object masks or scene graphs that enable higher-level reasoning, $R \in \{0, 1\}$.

This scoring mechanism is formalized as:

$$C_{\text{mod}} = \omega_1 M + \omega_2 Q + \omega_3 A + \omega_4 R, \qquad (9)$$

where modality sensitivity weights $\omega_i > 0$ for $i = 1, \dots, 4$ tune the relative importance of modality count, signal quality, temporal alignment, and reasoning-enabled annotations. For our benchmark, we set all weights equal to one.

Finally, to allow direct comparison across heterogeneous benchmarks, both raw scores are normalized. Task complexity to a standardized $[1, 5]$ scale and modality richness to a $[2, 5]$ scale. This mapping ensures interpretability: datasets with the lowest complexity or modality richness receive a score of 1 or 2 (Very Low/Minimal) and the highest receive 5 (Very High/Comprehensive), with intermediate values reflecting proportional positioning. The bubble size then encodes the relative dataset scale (e.g., number of episodes or hours), providing an at a glance summary of both range and comprehensiveness across the leading VLA benchmarks.

The resulting visualization effectively categorizes datasets, while it also highlights critical gaps in current benchmark, offering notably the underrepresented region combining highly complex tasks with extensive multimodal integration. This gap underscores a promising direction for future dataset development aimed at advancing truly generalist robotic agents capable of complex, real-world perception and planning.

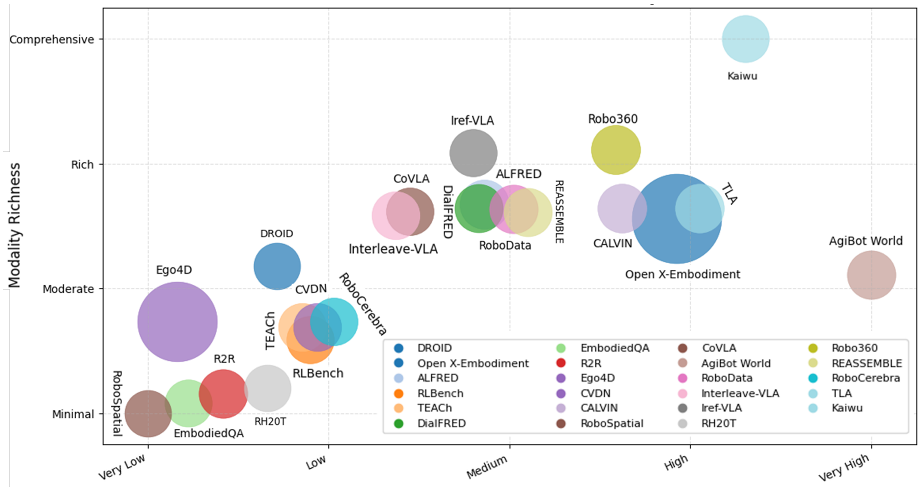


Figure 15: Benchmarking VLA Datasets by Task Complexity and Modality Richness. Each bubble represents a VLA dataset, positioned according to its normalized task-complexity score (x-axis) and its modality-richness score (y-axis). The bubble area is proportional to the dataset scale that is number of annotated episodes or interactions.

5.4. Benchmarking Analysis

Fig. 15 shows that most current VLA benchmarks are concentrated within a task complexity range from very low to high on the x-axis and span from minimal to rich modality along the y-axis. Early navigation and QA datasets like EmbodiedQA, R2R, and RoboSpatial are characterized by their very low complexity and minimal modality, reflecting simple, discrete decision-making in constrained environments. In contrast, mid-generation collections such as RLBench, TEACh, Ego4D, CVDN, and RoboCerebra, tend to feature low to moderate complexity with moderate modality richness, often focused on navigation, imitation, or basic manipulation tasks involving a limited number of modalities.

As the field evolves, datasets such as ALFRED, DialFRED, CoVLA, Interleave-VLA, RoboData, and REASSEMBLE have moved into the medium-complexity, rich-modality region by incorporating additional sensory streams like depth, language, and proprioceptive signals, enabling more sophisticated evaluation of policy learning and multistep planning. In particular, a small subset of datasets, including Iref-VLA, Robo360, TLA, CALVIN, and Open X-Embodiment, simultaneously achieve high task complexity and rich modality, each with a particular focus. Robo360 on multiview real-robot visual fidelity, Iref-VLA on referential grounding in 3D scenes, TLA on tactile-language-action alignment for contact-rich assembly, CALVIN on long-horizon language-conditioned robotic manipulation, and Open X-Embodiment on multirobot, multiskill demonstrations at scale.

The only dataset positioned at the extreme of both axes is Kaiwu, which achieves very high task complexity alongside the most comprehensive modality richness, integrating vision, depth, language, proprioception, haptics, and additional streams. Meanwhile, AgiBot World stands out in the very high complexity quadrant while exhibiting just moderate modality diversity, emphasizing large-scale, long-horizon dual-arm tasks rather than maximal sensor integration. This disparity highlights a critical gap: current VLA benchmarks do not yet fully integrate the challenges of long-horizon, multi-skill control with exhaustive multimodal input (vision, depth, language, proprioception, haptics, audio, and scene graphs). Without such datasets, the development of robust and generalist robotic agents remains limited. Future efforts should therefore focus on the upper right quadrant of the landscape, creating new VLA benchmarks that maximize both task difficulty and multimodal diversity to accelerate progress toward general-purpose embodied intelligence.

6. Simulation Tools

Simulation environments have become essential for VLA research, offering scalable, repeatable, and extensively annotated data at orders of magnitude greater than what is feasible in the physical world. Modern platforms such as AI2-THOR, Habitat, and NVIDIA Isaac Sim provide high-precision physics, realistic rendering, and customizable multimodal sensors ranging from RGBD cameras, force/torque and tactile probes, to proprioceptive encoders and language interfaces all configurable at fine temporal resolutions. Using procedural scene generation, randomized object properties, and scripted agent behaviors, simulators enable the automated synthesis of hundreds of thousands of trajectories, complete with ground truth annotations for object poses, semantic maps, action sequences, and natural language instructions. Crucially, built-in toolkits for scenario scripting and domain randomization facilitate systematic studies of generalization under varied lighting, object geometries, and task orders, while lightweight GPU accelerated backends support rapid iteration of new dataset designs. Together, this ecosystem of VLA simulators accelerates the co-development of control algorithms and benchmark datasets, ensuring that advances in multimodal perception, language grounding, and closed-loop planning can be evaluated and refined in a controlled, reproducible framework before deployment on real robotic platforms.

Table 3: Overview of simulation platforms commonly used for generating and evaluating VLA datasets. The table summarizes each simulator's supported sensory modalities, primary use cases, core capabilities, and the datasets that rely on them. These tools span diverse domains such as photorealistic indoor navigation, dexterous manipulation, and large-scale reinforcement learning, with varying degrees of physics realism.

Simulator	Modalities	Use Cases	Capabilities	Datasets
AI2-THOR (Kolve et al., 2017) **[link]**	RGB, depth, semantic/instance segmentation, object states	Embodied navigation, object manipulation	Photorealistic indoor scenes; procedural scene generation; physics-based object/agent interaction; built-in interaction APIs; language and task integration	ALFRED (Shridhar et al., 2020), TEACh (Padmakumar et al., 2021), DialFRED (Gao et al., 2022b)
Habitat (Savva et al., 2019) **[link]**	RGB, depth, semantic segmentation, agent pose	Vision-language navigation, embodied QA, point-goal navigation	Photorealistic, high-performance rendering; large-scale 3D scene support; modular sensor and agent APIs	R2R (Anderson et al., 2018), CVDN (Thomason et al., 2019), EmbodiedQA (Das et al., 2018)
NVIDIA Isaac Sim (NVIDIA Corporation) **[link]**	RGB, depth, LiDAR, semantic/instance segmentation, bounding boxes, point clouds, physics states, force/torque, event camera	RL & control, sim-to-real transfer, synthetic dataset generation, embodied AI, multi-robot simulation, digital twin, warehouse & industrial robotics	Physically accurate PhysX dynamics; RTX-accelerated photorealistic rendering; procedural scene generation; domain randomization; noise models; ROS/ROS2 and Python API; cloud deployment	Open X-Embodiment (Collaboration et al., 2025), Isaac Gym, RLBench, custom synthetic datasets
Gazebo (Koenig and Howard, 2004) **[link]**	RGB, depth, LiDAR, IMU, joint states, force/torque, contact, GPS	Control algorithm development, multirobot coordination, sim-to-real transfer, embodied navigation and manipulation	Open-source; plugin-based extensibility; realistic multi-physics engines; ROS1/ROS2 integration; multi-robot, multi-sensor support	RoboSpatial (Song et al., 2025a)
PyBullet (Coumans and Bai, 2016) **[link]**	RGB, depth, contact forces, joint states	RL, robotic manipulation prototyping, physics-based simulation	Real-time physics; Python API; cross-platform; easy scripting; supports robotics and VR	QUAR-VLA (Chen et al., 2024), various custom RL/-manipulation datasets
CoppeliaSim (Rohmer et al., 2013) **[link]**	RGB, depth, joint states, force/-torque sensors, proximity sensors	Multi-robot coordination, task scripting, manipulation, education	Multiple built-in physics engines; remote APIs (Python, ROS, C++); graphical scene editor; flexible scripting	RLBench (James et al., 2020), CALVIN (Mees et al., 2022)
Webots (Michel, 2004) **[link]**	RGB, depth, sound, GPS AI2-THOR (Kolve et al., 2017) **[link]**	RGB, depth, semantic/instance segmentation, object states	Embodied navigation, object manipulation Photorealistic indoor scenes; procedural scene generation; physics-based object/agent interaction; built-in interaction APIs; language and task integration	ALFRED (Shridhar et al., 2020), TEACh (Padmakumar et al., 2021), DialFRED (Gao et al., 2022b)
Habitat (Savva et al., 2019) **[link]**	RGB, depth, semantic segmentation, agent pose	Vision-language navigation, embodied QA, point-goal navigation	Photorealistic, high-performance rendering; large-scale 3D scene support; modular sensor and agent APIs	R2R (Anderson et al., 2018), CVDN (Thomason et al., 2019), EmbodiedQA (Das et al., 2018)
NVIDIA Isaac Sim (NVIDIA Corporation) **[link]**	RGB, depth, LiDAR, semantic/instance segmentation, bounding boxes, point clouds, physics states, force/torque, event camera	RL & control, sim-to-real transfer, synthetic dataset generation, embodied AI, multi-robot simulation, digital twin, warehouse & industrial robotics	Physically accurate PhysX dynamics; RTX-accelerated photorealistic rendering; procedural scene generation; domain randomization; noise models; ROS/ROS2 and Python API; cloud deployment	Open X-Embodiment (Collaboration et al., 2025), Isaac Gym, RLBench, custom synthetic datasets
Gazebo (Koenig and Howard, 2004) **[link]**	RGB, depth, LiDAR, IMU, joint states, force/torque, contact, GPS	Control algorithm development, multirobot coordination, sim-to-real transfer, embodied navigation and manipulation	Open-source; plugin-based extensibility; realistic multi-physics engines; ROS1/ROS2 integration; multi-robot, multi-sensor support	RoboSpatial (Song et al., 2025a),
PyBullet (Coumans and Bai, 2016) **[link]**	RGB, depth, contact forces, joint states	RL, robotic manipulation prototyping, physics-based simulation	Real-time physics; Python API; cross-platform; easy scripting; supports robotics and VR	QUAR-VLA (Chen et al., 2024), various custom RL/-manipulation datasets
CoppeliaSim (Rohmer et al., 2013) **[link]**	RGB, depth, joint states, force/-torque sensors, proximity sensors	Multi-robot coordination, task scripting, manipulation, education	Multiple built-in physics engines; remote APIs (Python, ROS, C++); graphical scene editor; flexible scripting	RLBench (James et al., 2020), CALVIN (Mees et al., 2022)
Webots (Michel, 2004) **[link]**	RGB, depth, sound, GPS, proximity, IMU, lidar, joint states	Mobile navigation, multi-robot and swarm robotics, manipulation, education	Cross-platform; extensive sensor and actuator models; GUI scenario/world design; ROS integration; physics-based simulation	AgiBot World (Bu et al., 2025)
Unity ML-Agents (Juliani et al., 2018) **[link]**	RGB, depth, raycasts, physics states	Reinforcement & imitation learning, interactive tasks	Unity engine visual fidelity; Python/C# APIs; curriculum learning	Used in custom RL and navigation datasets (e.g., Obstacle Tower, RoomNav, MiniWorld);
MuJoCo (Todorov et al., 2012) **[link]**	Joint positions, contact forces, kinematics, RGB	Continuous control, dynamics learning, RL research	High-speed simulation; accurate contact and soft body modeling; analytic gradients	Meta-World (Yu et al., 2020), RoboSuite (Zhu et al., 2020), custom RL benchmarks
iGibson (Xia et al., 2020) **[link]**	RGB, depth, semantic & instance masks, object poses, contact forces	Interactive navigation, manipulation, semantic reasoning	Photorealistic dynamic scenes; real-world scene reconstructions; interactive objects and agents	iGibson v1/v2 (Xia et al., 2020)
UniSim (Yang et al., 2023) **[link]**	RGB, depth, proprioception, haptics, audio	Multi-modal dataset generation, multi-agent coordination, manipulation, navigation	Unified multi-sensor API; scalable cloud-native simulation; plugin-based extensibility; support for real and simulated sensor data	UniSim-VLA (Yang et al., 2023)
SAPIEN (Xiang et al., 2020) **[link]**	RGB, depth, segmentation masks, contact forces, articulated object states	Deformable and articulated object manipulation, semantic reasoning, dexterous grasping	High-fidelity GPU-based physics; real-time dynamic simulation; Python API; large-scale articulated object library	DexGraspNet (Wang et al., 2023), TLA (Hao et al., 2025)

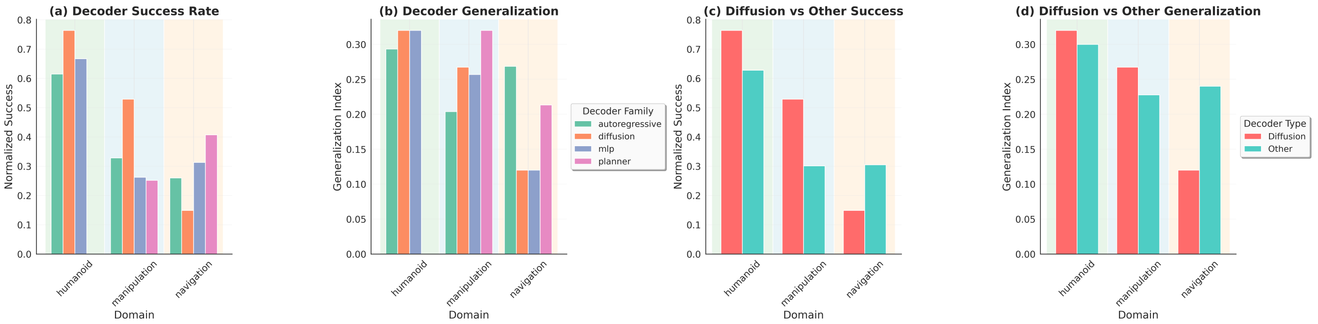


Figure 16: Cross-domain decoder performance analysis showing normalized success rate and generalization index across humanoid, manipulation, and navigation tasks for major decoder families; comparison of diffusion-based decoders versus all other types. Results demonstrate that diffusion decoders consistently achieve higher success and maintain competitive generalization across domains, highlighting their robustness for temporally coherent, cross-modal control.

Table 3 summarizes the current state-of-the-art simulation platforms used for the generation of VLA datasets. The table lists four essential aspects for each simulator: the *Modalities* of sensors it offers, the main *Use Cases* it supports, its fundamental technical *Capabilities*, and the representative *Datasets* that are based on it. This unified view allows researchers to directly compare engines based on their multi-modal sensor suite, physics accuracy, scalability, and integration with language and control APIs.

In the first column, platforms such as AI2-THOR and Habitat provide photorealistic RGB, depth, and semantic streams, making them ideal for embodied navigation and visual question answering benchmarks (e.g., ALFRED, R2R, EmbodiedQA, CVDN). Middle entries like NVIDIA Isaac Sim and Gazebo deliver advanced LiDAR, IMU, force/-torque, and multi-robot support crucial for large-scale reinforcement learning, sim-to-real transfer, and multi-agent coordination, as in Open X-Embodiment and RoboSpatial. Contact-rich simulators including PyBullet, CoppeliaSim, MuJoCo, and SAPIEN enable precise force, torque, and haptic feedback, powering dexterous manipulation datasets such as DexGraspNet, CALVIN, and TLA. Emerging platforms (Unity ML-Agents, RoboSuite, IsaacGym, UniSim) highlight capabilities such as GPU-parallel rollout, cloud-native simulation, and unified multi-sensor APIs, enabling the creation of next-generation VLA datasets with millions of diverse trajectories spanning vision, language, touch, and audio. The table provides an essential reference by mapping these four aspects across fifteen simulators: it assists in choosing the optimal backend for dataset generation, clarifies the trade-offs between rendering quality and processing speed, and identifies gaps where enhancements in simulator features could facilitate more detailed VLA benchmarks. In addition to Table 3, we also provide a capability matrix that compares sensor synchronization fidelity, physics realism, and multimodal annotation consistency across simulators to guide VLA dataset generation.

6.1. Simulator Capability Comparison

The suitability of a simulator for VLA data generation depends strongly on three practical dimensions. *Sensor synchronization* captures how reliably RGB, depth, LiDAR, proprioception, and force/torque streams are timestamped and aligned (frame-rate stability, cross-sensor jitter, and temporal drift). *Physics realism* reflects the accuracy of contact dynamics, friction, compliance, and articulated motion, critical for dexterous manipulation and sim-to-real transfer. *Multi-modal annotation consistency* assesses whether simulators can produce frame-synchronized logs (object poses, masks, actions, and language metadata) that remain aligned without drift over long sequences.

To support consistent comparison across platforms, we assign each simulator a qualitative rating, High (H), Medium (M), or Low (L) based on three criteria used throughout our survey: (i) published benchmarks and documentation on timing, physics, and export accuracy, (ii) empirical evidence from commonly used VLA and RL datasets generated in the simulator, and (iii) practical constraints reported in prior works (e.g., frame jitter, annotation drift, or inaccurate contacts). An “H” rating corresponds to strong alignment and high-fidelity performance with minimal drift or failure modes; “M” indicates adequate but imperfect support that may require custom tooling; and “L” denotes limited or inconsistent support for the given axis.

Different simulators excel in different regions of this space: photorealistic platforms (AI2-THOR, Habitat) benefit perception and language grounding but may be throughput-limited; contact-oriented engines (MuJoCo, SAPIEN) suit manipulation; multi-sensor/multi-robot platforms (Isaac Sim, Gazebo) support embodied integration and ROS pipelines; and massively parallel roll-out platforms (Isaac Gym, PyBullet) scale data generation but may widen domain gaps. The capability matrix below summarizes these trade-offs.

6.2. Limitations for large VLA data generation

Despite their strengths, current simulators exhibit several limitations for high-quality VLA dataset creation. *(i)*

Table 4
Simulator capability matrix. Columns denote: **Sync**: sensor synchronization fidelity; **Phys.**: physics realism; **Annot.**: multimodal annotation consistency; **Multi**: multi-robot support; **Notes**: key features for VLA data generation. Ratings: H = High, M = Medium, L = Low.

Sim	Sync	Phys.	Anno	Multi	Notes
AI2-THOR	M	M	M	M	Photorealistic scenes; strong for vision–language tasks; GPU-heavy.
Habitat	M	M	M	M	Fast rendering; large indoor scenes; VLN/VQA; export needs extra tools.
Isaac Sim	H	H	H	H	PhysX+RTX; rich sensors; ROS/ROS2; strong multi-robot and sim-to-real.
Gazebo	M	M	M	H	Open-source; broad sensors; strong multi-robot; annotation varies.
PyBullet	M	M	M	M	Fast physics; easy scripting; widely used for RL; moderate realism.
Coppelia	M	M	M	M	Multi-physics; remote APIs; basis for RLBench/CALVIN datasets.
MuJoCo	M	H	M	L–M	Accurate contacts; fast; external tools required for multimodal annotation.
SAPIEN	M	H	M–H	M	GPU physics; articulated objects; good for dexterity tasks.
Unity ML-Agents	M	M	M	M	High-fidelity visuals; flexible editor; curriculum learning.
RoboSui	M	M	M	M	Manipulation-focused; stable wrappers; export depends on configuration.
Isaac Gym	M	M	M	M	Massive GPU-parallel rollouts; high throughput; lower realism.
iGibson	M	M	M	L–M	Realistic reconstructions; interactive objects; export requires alignment.

Contact-rich physics: many engines rely on simplified friction and point-contact models, which lead to slip or unstable contacts in sim-to-real transfer. *(ii) Realism–throughput trade-offs:* photorealistic renderers improve visual grounding but reduce FPS, while lightweight backends scale rollouts but increase the domain gap. *(iii) Language grounding pipelines:* most platforms lack native instruction-to-behavior interfaces, requiring custom annotation toolchains that fragment formats. *(iv) Multi-robot variability:* URDF/SDF import, controller scheduling, and synchronization differ across platforms, complicating cross-robot pretraining and reproducibility.

These limitations echo the broader challenges summarized in Sec. 8.4 and highlight the need for unified, high-fidelity simulators tailored for large-scale VLA data generation.

7. Cross-Domain Evaluation and Analysis of VLA Models

This section presents a unified examination of how VLA models perform and generalize across various robotic domains. We begin by surveying the primary application areas where VLA architectures have demonstrated embodied intelligence, ranging from dexterous manipulation to mobile navigation and humanoid control, highlighting how architectural and training choices shape multimodal grounding efficiency. Building on this overview, we conduct a large-scale cross-domain benchmarking and meta-analysis, comparing encoder and decoder families in terms of normalized success and generalization.

We then introduce the *Vision Language Action Fusion Evaluation Benchmark (VLA-FEB)*, a standardized quantitative framework that integrates multimodal alignment, fusion energy, real-to-sim transfer, and generalization metrics to enable holistic performance assessment across architectures. A systematically compiled *Future Challenge Suite* is proposed to evaluate manipulation, navigation, reasoning, and sim-to-real robustness using tasks widely available in open-source simulators. Finally, a *Transparent Fusion Analysis* view links interpretability and causality to safety and reliability, showing how attention attribution, counterfactual testing, and entropy-based diagnostics can make embodied autonomy both explainable and trustworthy. Together, these components establish a comprehensive foundation for evaluating, comparing, and improving the next generation of general-purpose VLA systems.

7.1. Cross-Domain Meta-Analysis of VLA Performance

We extend our quantitative meta-analysis beyond manipulation to include VLA models evaluated in navigation and humanoid robotic tasks. This analysis provides a unified perspective on how architectural and dataset-level factors jointly influence normalized success rates and generalization indices across evaluated models. All metrics were standardized via min-max normalization within each benchmark family to ensure fair cross-domain comparison.

Fig. 16 analyzes component-level contributions. Diffusion decoders dominate across the humanoid, manipulation, and navigation domains, achieving both higher success and generalization than other families. This demonstrates that stochastic temporal modeling enhances action smoothness and multimodal grounding in continuous control. The details about the computation of the below plots are explained in Appendix D.

Fig. 17 further compares major decoder families across all domains combined. Diffusion-based decoders achieve the highest normalized success, outperforming planner, autoregressive, and MLP variants. In terms of generalization, diffusion and planner families show comparable indices, slightly above MLP and autoregressive models. These results confirm that while diffusion policies achieve greater success and stability in complex environments, simpler decoders can still generalize efficiently under constrained control settings. Overall, generative temporal models provide the most balanced trade-off between accuracy and cross-domain generalization.

Fig. 18 presents domain-specific results for both visual and linguistic encoders using simplified groupings. Among vision encoders, *SigLIP* leads in humanoid and manipulation domains, followed by *DINOv2* and hybrid transformer variants such as *Qwen Vision* in navigation. ResNet and CLIP-based encoders show moderate performance, indicating their effectiveness in spatial grounding but weaker adaptation

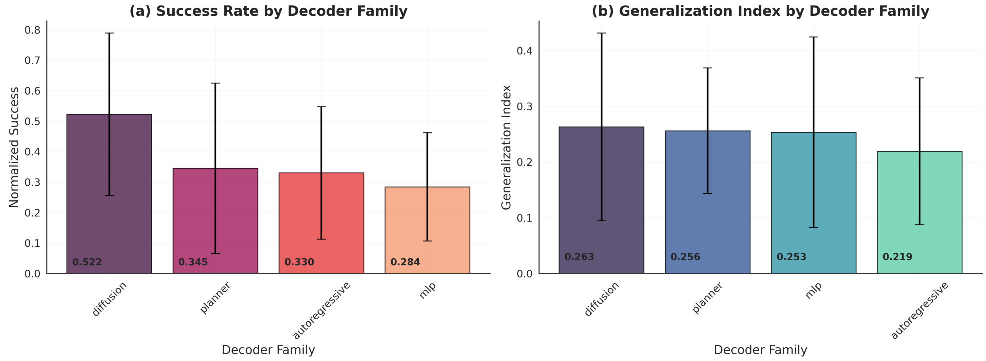


Figure 17: Decoder family comparison across all domains combined. Diffusion-based decoders achieve the highest success, while MLP and autoregressive models generalize most broadly.

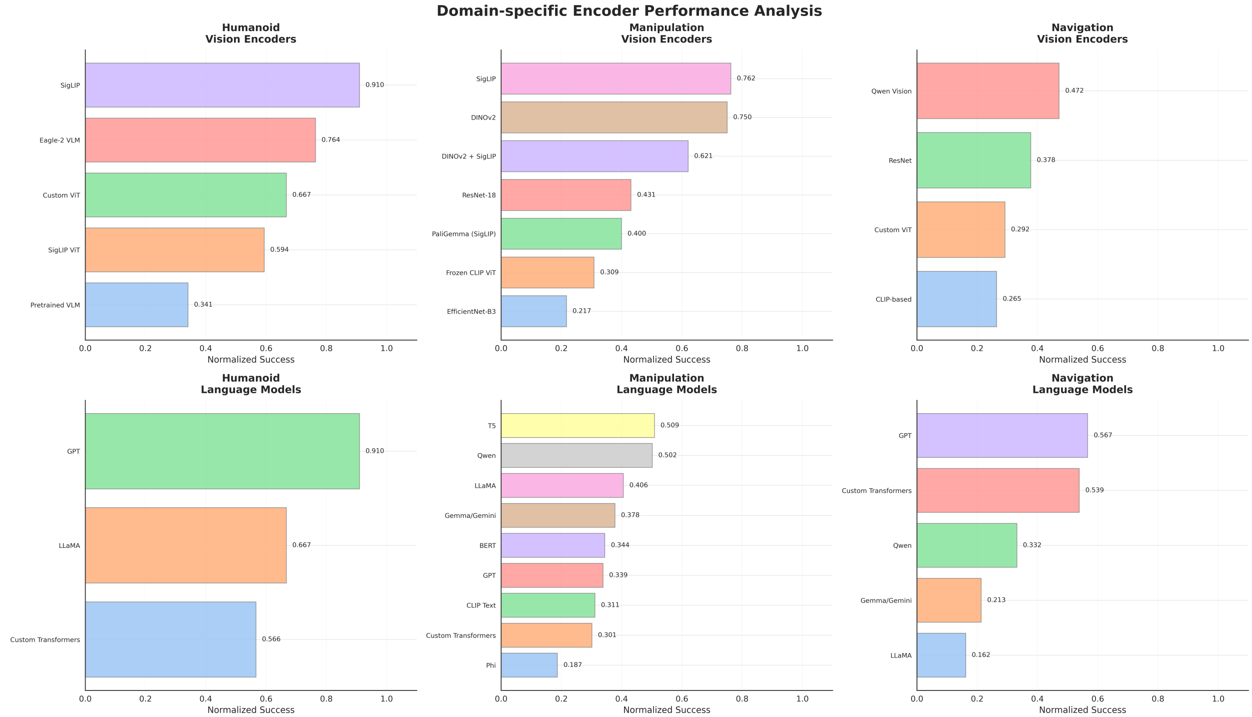


Figure 18: Domain-specific encoder performance analysis across humanoid, manipulation, and navigation tasks. Each subplot reports normalized success for vision and language encoder families. Transformer-based vision encoders (SigLIP, DINOv2, Qwen Vision) outperform convolutional baselines, while instruction-tuned language models (GPT, T5, Qwen, LLaMA) achieve superior multimodal grounding and cross-domain transfer.

to open-world settings. In general, transformer-based encoders demonstrate higher robustness and generalization across robot embodiments.

For language models, *GPT*-based encoders achieve the highest success in both humanoid and navigation domains, while *T5* and *Qwen* maintain strong manipulation performance. *LLaMA* and *Gemma/Gemini* families achieve balanced but slightly lower success, whereas compact or non-instruction-tuned models (*Phi*, *BERT*) remain less effective in multimodal grounding. These results indicate that instruction tuning and model scale, rather than size alone, are crucial for effective cross-domain generalization.

Fig. 19 provides a comprehensive comparison of vision and language encoders by combining all domains. Among visual encoders, *SigLIP* achieves the highest success, followed by *LLaVA/VILA* and *Qwen-VL*. Although *CLIP* and *EfficientNet* yield lower success, they maintain moderate generalization. This pattern indicates that transformer-convolutional hybrids balance spatial precision with semantic abstraction. For language models, *T5/Flan* and *GPT*-family encoders achieve the highest success, while *Gemma* and *LLaMA* show slightly lower success but stronger generalization. Compact instruction-tuned LLMs such as *Qwen* also demonstrate solid trade-offs between efficiency and transferability, outperforming smaller text-only encoders (*CLIP*, *Phi*).

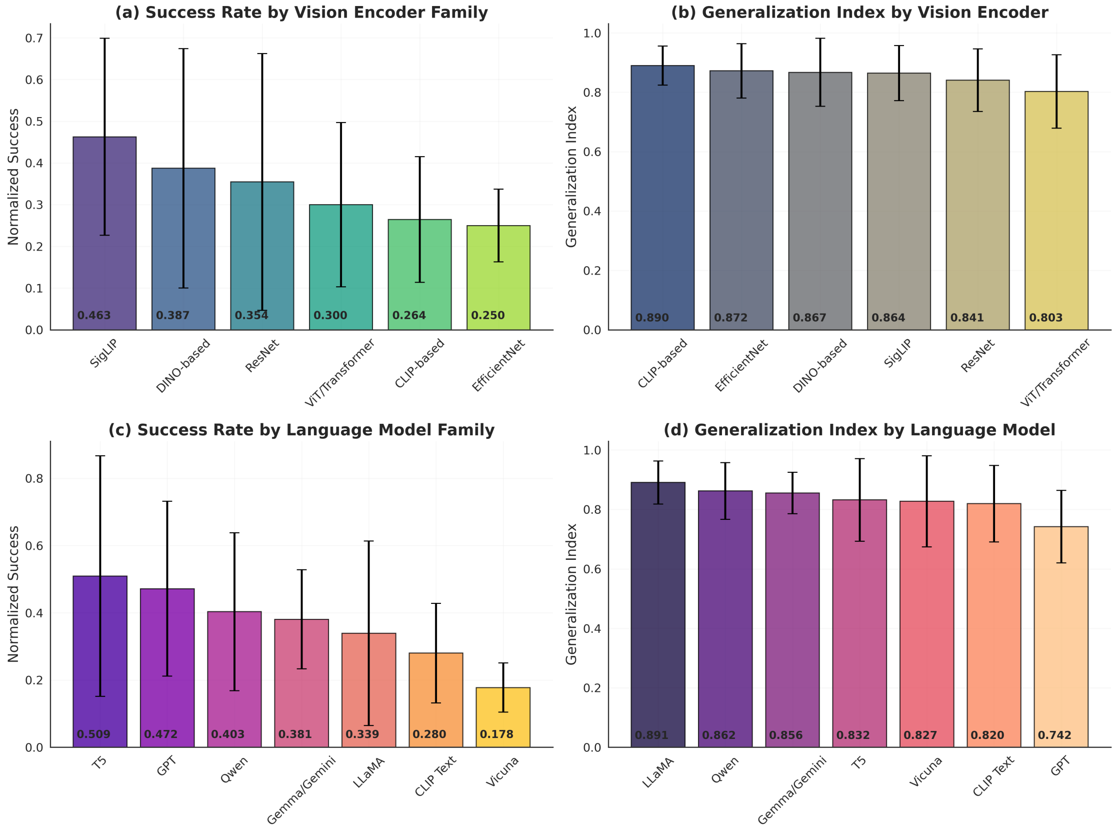


Figure 19: Vision and language encoder family performance across all domains combined: success and generalization for visual backbones and corresponding trends for language models. Results indicate that SigLIP, DINO, and mid-scale instruction-tuned language models (T5, LLaMA, Qwen) provide the best balance between task success and generalization.

Collectively, these findings reveal several convergence-level insights. Diffusion-based decoders dominate across domains due to their temporal coherence and stochastic grounding. Transformer-derived vision encoders (*SigLIP*, *DINO*, *ResNet*) provide stable perceptual grounding, while convolutional variants remain useful for local spatial precision. Mid-scale, instruction-tuned language models (*T5*, *LLaMA*, *Qwen*) generalize effectively without extreme parameter scaling.

This cross-domain benchmarking highlights consistent trends across humanoid, manipulation, and navigation tasks: diffusion-based decoders and transformer-driven encoders achieve the most stable and generalizable performance. While CNN backbones enhance spatial reasoning, transformer architectures dominate embodied perception, and medium-scale instruction-tuned LLMs achieve optimal efficiency and generalization balance. These results collectively point toward a convergent trajectory in multimodal robotics, where balanced architectures, generative temporal decoders, and hierarchical fusion jointly drive robust, transferable embodied intelligence.

7.2. VLA-FEB: Unified Benchmarking Framework

To unify the evaluation of multimodal fusion in robotic systems, we introduce the *Vision Language Action Fusion Evaluation Benchmark (VLA-FEB)*, a standardized protocol that advances beyond qualitative comparison and establishes quantitative metrics capturing the quality, efficiency, and transferability of fusion processes. The benchmark evaluates four complementary dimensions that jointly characterize a model's ability to ground perception and language into effective action. The first component, the Cross-Modal Alignment Score (CMAS), quantifies how consistently visual and linguistic embeddings remain aligned during task execution:

$$\mathrm{CMAS} = \mathbb{E}\big[\cos\big(f_V, f_L\big)\big], \tag{10}$$

where f_V and f_L denote latent visual and linguistic representations extracted at each action step. A high CMAS indicates that perceptual and instructional streams remain semantically synchronized as the robot interacts with the environment.

The second component, the Fusion Energy Index (FEI), approximates the theoretical E_{fusion} (Sec. 4.5) through measurable information-theoretic quantities:

$$\mathrm{FEI} = \sum_k \eta_k \, \Delta H_k, \tag{11}$$

where ΔH_k represents entropy reduction at fusion layer k, and η_k is its normalized efficiency factor. This metric

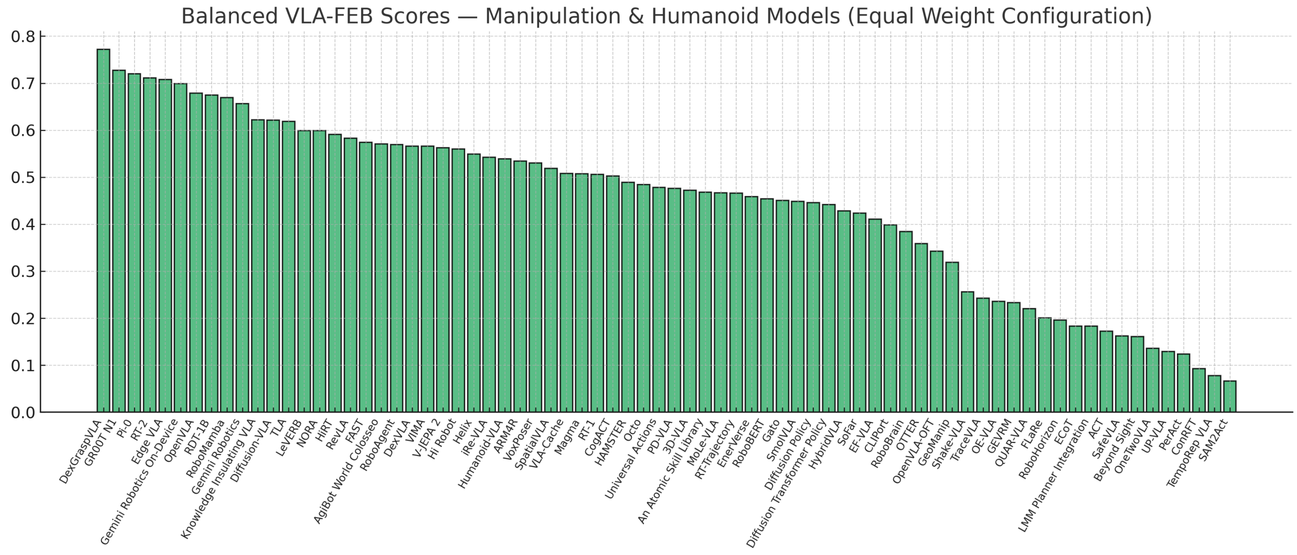


Figure 20: Normalized VLA-FEB composite scores across evaluated architectures: Each bar represents the aggregated performance of a model under equal weighting of fusion efficiency, generalization, real-to-sim transfer, and cross-modal alignment (w_{fusion}=w_{GI}=w_{R2S}=w_{CMAS}=0.25). Hierarchical and diffusion-based models achieve the highest composite success, indicating that architectures integrating semantic reasoning with probabilistic control deliver the most balanced and transferable performance across domains.

reflects how effectively multimodal integration reduces uncertainty in the robot's policy distribution.

The Real-to-Sim Transfer Efficiency (R2S) measures the fidelity of simulated pre-training to real-world deployment:

$$\text{R2S} = \frac{S_{\text{real}}}{S_{\text{sim}}}, \tag{12}$$

where S_{real} and S_{sim} denote normalized task success rates in real and simulated conditions, respectively. Higher R2S values indicate stronger sim-to-real generalization and policy robustness under real-world uncertainty.

The Generalization Index (GI) evaluates a model's stability when encountering unseen tasks:

$$\text{GI} = 1 - \frac{\sigma_S}{\bar{S}}, \tag{13}$$

which measures performance consistency across new task-object pairs, penalizing models that exhibit large variance in success rates.

Collectively, these metrics form a benchmark-driven evaluation framework that quantifies not only *whether* a model succeeds, but also *how* and *why* it succeeds. A composite VLA-FEB score enables a unified ranking across architectures:

$$\text{VLA-FEB} = w_1\,\text{CMAS}{+}w_2\,E_{\text{fusion}}{+}w_3\,\text{R2S}{+}w_4\,\text{GI}, \tag{14}$$

with tunable weights w_i reflecting emphasis on semantic grounding, efficiency, and transferability. For the final configuration, all weights are assigned equally as w_{fusion}=w_{GI}=w_{R2S}=w_{CMAS}=0.25, ensuring balanced consideration of multimodal alignment, fusion efficiency, generalization capability, and real-to-sim transfer. This uniform weighting highlights models that achieve a holistic equilibrium across all four evaluation dimensions rather than excelling in a single criterion.

Fig. 20 further presents the normalized composite VLA-FEB scores of all evaluated architectures under equal weighting (w_{fusion}=w_{GI}=w_{R2S}=w_{CMAS}=0.25). Hierarchical and diffusion-based models such as *DexGraspVLA*, *GR00T-N1*, and *Pi-0* consistently outperform early- and late-fusion baselines, confirming that deeper multimodal integration and probabilistic policy decoders jointly enhance generalization, robustness, and sim-to-real transfer. Mid-tier architectures such as *RT-2* and *OpenVLA* perform competitively due to strong pretraining alignment, whereas purely symbolic or shallow-fusion systems show significant declines in composite performance. This distribution quantitatively validates the theoretical framework established in Section 4.5, demonstrating that multimodal synergy, rather than scale alone, serves as the primary driver of embodied generalization.

7.3. Future Challenge Suite for Embodied Evaluation

Several open-source embodied simulators and task suites, such as RLBench, ManiSkill, ALFRED, and Habitat, already provide diverse multimodal environments for evaluating perception, language, and control integration. Building upon these foundations, we propose a selected *Future Challenge Suite* (Table 5) comprising representative tasks that are widely supported across existing simulators. The selected tasks span manipulation, navigation, embodied reasoning, and sim-to-real transfer scenarios, enabling systematic assessment of distinct aspects of VLA model capability, such as architectural fusion efficiency, cross-modal generalization, temporal grounding, and embodiment robustness.

Table 5
Representative task prompts from the Future Challenge Suite demonstrating multimodal grounding and fusion complexity.

Task Group	Example Task Prompts	Fusion / Challenge
A. Language-Guided Manipulation		
Pick-and-Place Variants	Pick the red cube and place it on the blue platform. Lift the green cup and put it on the tray.	Vision-language grounding
Stacking / Sorting	Stack three yellow blocks from small to large. Sort fruits by color into bowls.	Temporal grounding
Tool Use (Hammer, Scoop)	Use the hammer to hit the nail. Scoop sand and pour into the bucket.	Causal fusion
Cable Insertion / Knot-Tying	Insert the plug into the port. Tie a knot with the rope.	Tactile-visual synergy
Assembly (Nut-Bolt, Peg-in-Hole)	Fasten the bolt to the nut. Insert the peg into the hole.	Precision fusion
B. Goal-Conditioned Navigation		
Point-Goal Navigation	Go to the red chair near the window. Move to the charging station.	Spatial grounding
Dynamic Obstacle Avoidance	Reach the door while avoiding moving boxes.	Attention stability
Semantic Map Following	Follow the corridor to the kitchen.	Semantic mapping
Multi-Floor Navigation	Take the elevator to floor 2 and find room A.	3-D reasoning
Long-Horizon Search	Locate the buoy marked B in the harbor.	Policy generalization
C. Embodied Reasoning and Planning		
Referring Expression Grounding	Find the cup next to the red book.	Object reference
Compositional Task Parsing	Pick the spoon, place it in the cup, then push it.	Sequence fusion
Conditional Logic Execution	If the door is open, enter; otherwise, knock.	Logical policy
Causal Chain Planning	Press the switch, then the green button.	Temporal causality
Open-World Embodied QA	What object remains after removing the pen?	State reasoning
D. Perception-Language-Action Generalization		
Color / Shape Variation	Pick the smallest blue cube.	Feature alignment
Unseen Object Grasping	Grasp the unfamiliar tool near the wrench.	Domain generalization
Novel Tool / Scene Composition	Use the new screwdriver on the hinge.	Representation reuse
Cross-Embodiment Execution	Repeat the task using the humanoid arm.	Embodiment invariance
Multi-Agent Cooperation	Coordinate with the drone to lift the box.	Cooperative fusion
E. Sim-to-Real Transfer and Robustness		
Pose Imitation / Retargeting	Mimic the demonstrated human arm motion.	Visual-proprio fusion
Visual Domain Randomization	Perform pick-and-place under lighting changes.	Noise robustness
Dexterous Manipulation (Real)	Rotate the valve until it reaches 90 degree.	Tactile fusion
Outdoor Mobile Manipulation	Deliver the tool to the worker outside.	R2S transfer
Adaptive Multi-Robot Sharing	Collaborate with the quadruped to move the beam.	Multi-agent fusion

Each task category in the suite is designed to examine a specific dimension of multimodal intelligence. Manipulation tasks assess how effectively the system fuses visual and linguistic cues to achieve precise object interactions and tool use under varying spatial and causal conditions. Navigation tasks evaluate spatial reasoning, goal-directed planning, and attention stability when following language instructions in dynamic environments. Embodied reasoning tasks challenge compositional understanding and causal inference, testing whether models can parse sequential or conditional instructions and execute them systematically. Finally, sim-to-real and robustness tasks measure the transferability of learned policies from simulation to the physical world, examining visual domain adaptation, proprioceptive feedback integration, and stability under uncertainty. These tasks enable a holistic evaluation of how well VLA systems align perception, language, and action, capturing their strengths and weaknesses in real-world generalization, adaptive control, and multimodal fusion. The *Future Challenge Suite* thus serves as a unified and extensible benchmark for the next generation of embodied intelligence systems.

7.4. Transparent Fusion Analysis and Interpretability

To ensure that multimodal fusion in VLA systems is not only effective but also understandable and trustworthy, this section introduces a causal and interpretability perspective. While Sec. 4.4 and 7.2 quantified fusion efficiency and generalization, here we examine *why* and *how* different modalities influence decision making in embodied robotic systems.

1) Causal Reasoning in Fusion: VLA models integrate vision, language, and proprioception through cross-modal attention and diffusion-based decoding. Understanding causal dependencies among these modalities is essential for safe autonomy. A simplified causal view can be expressed as:

$$V, L, S \rightarrow Z_k \rightarrow A, \tag{15}$$

where V, L, and S represent vision, language, and state inputs; Z_k denotes the latent representation at fusion layer k; and A is the resulting action. By performing *interventions* for instance, masking visual inputs or modifying linguistic tokens, we can observe how changes in each modality causally affect the predicted action distribution $p(A|Z_k)$. This directly connects with the entropy-reduction framework in Sec. 4.5, where stronger causal alignment corresponds to greater uncertainty reduction (ΔH_k) and higher fusion efficiency (η_k).

2) Attention Attribution and Representation Transparency: Attention maps and token-level importance weights offer a concrete view of what the model attends to when generating actions. During manipulation, attention typically concentrates on grasp points or object surfaces, while instruction tokens such as “place” or “stack” activate semantic channels in the decoder. Visualizing these attention distributions helps verify whether decisions rely on physically meaningful cues rather than spurious correlations, providing a diagnostic lens for identifying misalignment between perception and language grounding.

3) Counterfactual and Latent Probing: Counterfactual testing asks “what-if” questions such as: *What if the instruction changes but the image remains constant? What if the object’s color or shape is altered?* Such analysis tests whether the latent representation Z_k captures genuine causal relations or superficial correlations. As illustrated in Fig. 12, the latent-factor analysis shows that hierarchical fusion layers frequently disentangle semantic intent (from language) based on geometric affordances (from vision), confirming that structured fusion supports more interpretable and causally organized internal representations.

4) Safe and Explainable Autonomy: In embodied robotics, interpretability is not merely descriptive; it forms the analytical bridge between quantitative fusion metrics and real-world reliability. The causal and attention-based signals identified above serve as measurable indicators of system stability. For example, sudden decreases in entropy reduction (ΔH_k) or fusion efficiency (η_k) across layers may indicate inconsistent reasoning flow, while divergences between expected and observed attention distributions can expose perception-action mismatches. Monitoring these indicators in real time transforms interpretability into a proactive safety mechanism: it allows early detection of unstable behaviors, ambiguous command responses, or perception drift. By linking interpretability metrics with theoretical constructs (Sec. 4.5) and benchmark outcomes (Sec. 7.2), this framework operationalizes transparency as a quantifiable dimension of safety and robustness in embodied autonomy.

Causality-based interpretability complements the quantitative evaluation introduced earlier by revealing the internal reasoning dynamics of VLA models. Integrating such *transparent fusion analysis* into VLA design unifies performance assessment with accountability, advancing safe, explainable, and verifiable robotic intelligence.

8. Progress, Challenges and Future Directions

This section synthesizes current progress, persistent limitations, and emerging research directions in VLA models. While recent advances demonstrate rapid improvements in multimodal grounding, architectural integration, and generalization across tasks and embodiments, several fundamental challenges continue to constrain the deployment of VLA systems in real-world environments. We organize these issues into three interconnected areas, architectural, dataset, and simulation challenges, and outline future directions that integrate modality-aware tokenizers, dynamic fusion, scalable multimodal datasets, and high-fidelity simulation tools. Together, these insights provide a roadmap to build robust, generalizable, and transparent VLA-based robotic autonomy.

8.1. Progress

Recent surveys converge on the view that VLA models have shifted from proof-of-concept systems toward generalist, deployable policies that jointly perceive, reason, and act across diverse tasks and embodiments. These reviews document steady gains in (i) architectural unification of perception, language grounding, and control, (ii) parameter and data-efficient adaptation, and (iii) training/evaluation pipelines spanning simulation and real robots (Sapkota et al., 2025; Kawaharazuka et al., 2025; Zhong et al., 2025a). A key conceptual advance is the explicit *action-token* view, which organizes policies by how actions are represented (e.g., language instructions, code, affordances, trajectories, latent actions, raw motor commands, or goal states) and how these tokens are produced and consumed along the perception→planning→control stack (Zhong et al., 2025a). This perspective helps explain why VLA policies with identical vision/language backbones can differ markedly in real-time control, safety, and generalization.

Despite rapid progress, the literature consistently highlights open problems: (1) long-horizon, partially observed control and memory; (2) alignment between high-level semantics and low-level continuous actions (closing the spatial/temporal gap); (3) sim-to-real robustness and cross-embodiment transfer at scale; (4) safety, interpretability, and evaluation standardization across tasks and datasets; and (5) efficiency constraints for on-robot inference (Sapkota et al., 2025; Kawaharazuka et al., 2025). Work that treats action representation as the primary design variable (via action tokenization) further reveals trade-offs among temporal precision, latency, safety verification, and data needs (Zhong et al., 2025a). These insights motivate standardized scoring protocols and causal/attribution tools for transparent fusion analysis (cf. our VLA-FEB design and interpretability metrics in Sec. 7.2 and Sec. 4.5).

These sources reinforce our framing that fusion depth, encoder scale, decoder class, and dataset coverage jointly drive success and generalization. They also support our call for a *benchmark-driven* protocol (VLA-FEB) linking alignment, fusion benefit, generalization, and deployment reliability with transparent, task-factorized reporting (Sec. 7.1, Sec. 7.2). Together, they provide a broader cross-domain rationale for our quantitative analysis and proposed evaluation suite.

8.2. Architectural Challenges

VLA models rely on a unified Transformer backbone to process high-resolution images or video frames alongside natural-language instructions and output platform-specific action commands. This end-to-end approach exposes several core architectural challenges that arise from the heterogeneity, scale, and physical diversity inherent to robotic control.

1. Tokenization and Vocabulary Alignment: VLA models must process heterogeneous inputs including natural language, image patches, and continuous robot states, however standard techniques such as byte-pair encoding (BPE) for text and fixed patch embeddings for vision often fail to capture the complexities of visual and proprioceptive signals. This misalignment results in inconsistent token distributions and degraded cross-modal attention. To address this, recent approaches have introduced unified tokenization schemes. Perceiver IO uses shared latent arrays for multimodal fusion (Jaegle et al., 2022), BLIP-2 introduces a Q-former to dynamically select vision tokens compatible with language models (Li et al., 2023), and adapter-based quantization layers allow flexible discretization within each modality stream (Pfeiffer et al., 2020). Despite these advances, several key challenges remain, such as efficiently encoding high-dimensional sensor streams without information loss, dynamically adapting vocabularies in the presence of noise or novel configurations, achieving low-latency token generation on resource-constrained platforms, and designing interpretable token spaces to support transparent and reliable cross-modal reasoning.

2. Modality Fusion: Simply concatenating visual and linguistic features or applying basic cross-attention often fails to align the distinct statistical properties of pixel-level and word-level representations, resulting in weak visual grounding. Recent advances adopt an "align-then-fuse" paradigm to strengthen cross-modal representations. For instance, align-before-fusing employs momentum-based contrastive learning to pre-align vision and language modalities (Li et al., 2022), and VLMo introduces multimodal expert layers within Transformer blocks to adaptively balance contributions from each stream (Wang et al., 2022). Despite these gains, key challenges remain: effectively fusing asynchronous sensory streams like haptics or audio; incorporating additional modalities such as force/torque signals; dynamically reweighting modality importance under domain shift (e.g., lighting changes or ambiguous language); improving interpretability of cross-attention layers for debugging; and enabling low-latency, resource efficient fusion for deployment on embedded robotic platforms.

3. Generalization Across Embodiments: Fixed action vocabularies and rigid kinematic bindings severely limit the ability of VLA models to transfer across different robot models. Recent approaches address this by conditioning action generation on robot-specific descriptors or learned affordance models. For example, PaLM-E encodes explicit hardware embeddings to adapt vision-language reasoning to new platforms (Driess et al., 2023), while RT-2 freezes its vision-language planning module and delegates embodiment/model-specific control to a lightweight action adapter. More recent efforts, such as DexVLA, go further by enabling plug-and-play cross-embodiment adaptation using diffusion-based expert modules trained across diverse kinematic structures. Despite these advances, zero-shot generalization to entirely novel robot models, payload distributions, or joint limits continues to degrade without fine-tuning. Moreover, sim-to-real transfer remains unstable under noisy sensor readings and unexpected dynamics, and generating smooth, compliant trajectories that adapt to varying torque, speed, and stiffness profiles across platforms remains an open and critical challenge.

4. Manipulator Motion Smoothness: Although many VLA models emphasize the prediction of discrete action tokens, they often neglect the quality of continuous motion trajectories, which are essential for smooth, safe and precise manipulation. Recent approaches such as Diffusion Policy (Chi et al., 2023b) reformulate visuomotor control as a conditional denoising process, enabling the generation of temporally coherent action sequences. Based on this, the diffusion transformer policy (Hou et al., 2024b) integrates large transformer architectures directly into the diffusion framework, achieving improved stability and generalization across diverse robotic platforms. However, several challenges remain unresolved: achieving real-time inference with latency-sensitive diffusion models, ensuring robust collision avoidance under sensor noise and dynamic uncertainty, maintaining a balance between trajectory smoothness and fast reactivity to changing goals, and coupling diffusion-based controllers with high-level language planners.

8.3. Dataset Challenges

Comprehensive, varied, and well-organized datasets form the basis for developing VLA models. However, current data sets exhibit several significant limitations that obstruct the path toward robust, general-purpose VLA models.

1. Task Diversity: Current datasets are highly specialized, focusing on narrow, short-horizon tasks. For instance, ALFRED and CALVIN emphasize pick-and-place operations, while R2R focuses on navigation and finding pathways guided by language. However, few datasets integrate long-horizon task planning that combines spatial reasoning, navigation, and fine-grained object manipulation in open-ended, multi-scene environments. This fragmentation restrains the training of agents capable of seamlessly switching between locomotion and manipulation tasks in realistic household or industrial scenarios.

2. Modality Imbalance: Most VLA datasets primarily offer RGB images and textual annotations, often excluding critical sensor modalities such as depth maps, force/torque signals, tactile feedback, or proprioceptive data. When these streams are present, they are frequently captured at inconsistent sampling rates or resolutions. This lack of high-quality, synchronized multimodal data significantly limits the development of models that can perform robust sensor fusion despite environmental uncertainty.

3. Annotation Quality and Cost: Obtaining accurate labels such as 6-DoF object poses, frame-aligned multi-sensor data, or detailed natural language explanations is resource intensive and time-consuming, requiring either detailed manual annotation or unreliable semi-automated pipelines. Although simulated environments can provide perfect annotations at scale, domain gaps in appearance, physics, and interaction fidelity often degrade sim-to-real transfer. Meanwhile, current self-supervised and auto-labeling methods remain unreliable across diverse task domains.

4. Realism and Scale: Real-world datasets like Open X-Embodiment offer high fidelity data with authentic sensor noise and physical interactions, but are constrained by the cost and time of robot data collection, typically producing only hundreds of hours of recordings. In contrast, simulation platforms can generate millions of trajectories efficiently but struggle to replicate complex real-world dynamics, such as material deformation, lighting variability, or occlusion effects. This trade-off between realism and scalability remains a fundamental bottleneck in the development of models that generalize beyond laboratory conditions.

Addressing these limitations will require coordinated efforts to build long-horizon, cross-domain benchmarks; gather richly synchronized multimodal datasets; reduce annotation costs through self-supervision and automation; and bridge the realism-scale divide via hybrid simulation-real data pipelines. These advances are essential to equip future VLA models with the robustness and adaptability needed for deployment in real-world environments.

8.4. Simulation Challenges

Simulators provide scalable, controllable environments for generating training data for VLA models. However, several critical limitations must be addressed to ensure that simulated performance reliably transfers to real-world deployment.

1. Physics Accuracy and Contact Modeling: Popular physics engines such as MuJoCo, PyBullet, and NVIDIA Isaac Sim simplify physical interactions by relying on basic Coulomb friction models and point-contact approximations. Although this enables stable and fast simulation, it fails to capture essential dynamics such as soft-body deformation, variable surface friction, and joint compliance. As a result, policies trained in simulation often perform poorly in the real world, leading to issues like object slip, unexpected torque spikes, or unstable contact behavior.

2. Visual Realism and Throughput Trade-offs: High-fidelity simulation platforms such as AI2-THOR, Habitat, and Unity ML-Agents provide photorealistic rendering and diverse assets, making them ideal for vision-heavy tasks. However, this comes at the cost of low frame rates and high GPU demand, limiting their suitability for large-scale reinforcement learning or self-supervised pretraining. In contrast, lightweight renderers support high-throughput simulation but suffer from domain gaps in texture, lighting, and occlusion realism, reducing the effectiveness of domain-randomized policies during real-world deployment.

3. Lack of Built-in Language Grounding APIs: Most simulators do not provide native support for grounding natural language commands into agent behaviors. This forces to create custom annotation pipelines such as those used in ALFRED or TEACh that align textual instructions with actions and scene representations. These efforts introduce significant development overhead, restrict reproducibility, and lead to fragmented and non-standardized data formats.

4. Multi-Robot and Agent Support Capabilities: Support for multiple robots varies widely across simulators. Some platforms like Isaac Sim and Gazebo offer flexible import of arbitrary robot descriptions via URDF or SDF formats, facilitating multi-robot coordination and benchmarking. Others, like Webots and RoboSuite, are optimized for specific robot families, limiting generalization and reusability. This inconsistency complicates cross-platform pretraining and impairs reproducibility across hardware setups.

Overcoming these challenges requires advancing contact-rich physics modeling, optimizing rendering pipelines for both fidelity and throughput, developing standardized language grounding interfaces, and unifying multi-agent simulation support. These improvements are essential to create simulation platforms that can produce realistic, scalable, and transferable datasets for training generalizable VLA models.

8.5. Safety, Failure, and Robustness Evaluation

While VLA systems demonstrate strong generalization across diverse manipulation and navigation tasks, safety and robustness remain major open challenges, particularly when operating in cluttered or uncertain environments. Unlike classical control pipelines with explicit stability guarantees, VLA policies rely on learned multimodal representations whose failure modes are often opaque, difficult to predict, and hard to diagnose.

Safety-critical failure modes in VLA-controlled manipulation typically arise from: (i) misaligned visual attention leading to incorrect object selection, (ii) ambiguity in language instructions producing wrong subgoals, (iii) distribution shifts in lighting, occlusion, or scene layout, and (iv) instability in contact-rich interactions such as grasping, pushing, and tool use. These errors may propagate through the perception-language-action loop, resulting in unsafe behaviors such as collisions, dropped objects, excessive forces, or unreachable configurations.

Uncertainty quantification remains an underdeveloped area across existing VLA frameworks. Most systems do not model epistemic or aleatoric uncertainty, nor do they estimate confidence over visual grounding, subgoal selection, or action generation. This makes it difficult to detect early-stage failures or trigger safe fallback behaviors. Incorporating uncertainty-aware components, such as Bayesian action decoders, ensemble critics, confidence-aware grounding modules, or diffusion models with calibrated likelihoods offers a promising direction for robust embodied autonomy.

Risk-aware planning is also insufficiently explored in current VLA architectures. While many systems optimize for task completion, few explicitly reason about hazardous states, safe regions, collision proximity, or force limits. Integrating VLA models with safety filters, control-barrier functions, model-predictive safety layers, or agentic verification loops could enable safer execution, especially in dense manipulation scenes.

failure diagnosis and interpretability remain crucial. Although recent VLA frameworks incorporate attention maps or vision-language introspection, these tools offer only partial insight into why failures occur. Future work should combine multimodal saliency, causal attribution, and trajectory-level diagnostics to create transparent post-hoc explanations and actionable feedback for system improvement. Addressing safety, uncertainty, and robustness is essential for moving VLA-driven manipulation from controlled labs into real-world, safety-critical environments.

8.6. Real-Time Constraints in VLA Systems

Although VLA models demonstrate remarkable capabilities in multimodal reasoning and manipulation, their practical deployment on physical robots remains constrained by real-time performance requirements. In contrast to offline simulation or batch inference scenarios, embodied systems are required to function within tight constraints on timing, energy consumption, and computational resources. These constraints directly influence the safety, feasibility, and robustness of VLA-driven control.

A primary challenge is *inference latency*. Many state-of-the-art VLA architectures rely on deep transformer encoders and diffusion-based decoders, which commonly require tens to hundreds of milliseconds per prediction step, far slower than the 10-100 Hz control loops needed for dexterous manipulation and fast closed-loop correction. Models with long multimodal token sequences and cross-attention layers further exacerbate this bottleneck. Although recent approaches introduce more efficient formulations, such as Mamba-based linear-time transformers (e.g., RoboMamba), these improvements remain insufficient for high-frequency real-world deployment.

Another major limitation is *memory and GPU footprint*. Large vision encoders, such as DINOv2 ViT-G, CLIP ViT-L/14, and the large visual backbone architectures employed in Octo and OpenVLA, impose heavy memory requirements (around 8-40 GB of GPU Memroy), making on-device deployment challenging. This creates performance gaps when deploying VLAs in embedded systems where power consumption and thermal limits are restrictive. To address these issues, recent work such as EdgeVLA and Gemini-on-Device emphasises lightweight visual language fusion and memory efficient inference pipelines designed specifically for on-device or low-power execution. These efforts demonstrate promising progress, but their performance remains significantly lower than that of full-scale VLA architectures.

Computational bottlenecks also arise from multimodal fusion itself. Cross-attention layers linking vision, language, and proprioception tokens introduce quadratic complexity with respect to sequence length. Diffusion-based action decoders require iterative denoising steps, increasing inference time linearly with the number of sampling iterations. Some recent architectures mitigate these limitations via token pruning, low-rank adaptation (LoRA), Mixture-ofthrough token pruning, low-rank adaptation (LoRA), Mixture-of-Experts (MoE) routing, frequency-space action tokenisation, or hybrid diffusion–autoregressive optimizations across VLA families is still lacking.

In general, despite the rapid progress of VLA research, the field lacks a unified understanding of how inference latency, memory consumption, power constraints, and architectural design choices jointly affect real-time robotic deployment. Addressing these gaps will require dedicated benchmarks, standardized latency and power reporting, and the development of resource-efficient VLA architectures capable of robust control under the computational conditions of real-world robots.

8.7. Future Directions

8.7.1. Unified Roadmap for Multimodal VLA Systems

To advance the next generation of VLA models, future systems should incorporate learnable, modality-aware tokenizers-such as vector-quantized VAEs or neural dictionaries to jointly discretize continuous sensor streams like proprioception and force/torque alongside visual and textual inputs. Dynamic fusion blocks (e.g., gating networks, mixture-of-experts, or conditional attention) can reweight each modality based on task demands, improving flexibility and robustness. For scaling long video or text sequences, hierarchical architectures are recommended, where lightweight CNN or RNN frontends downsample high-frame-rate inputs before passing them to sparse Transformer layers for efficient long-range modeling. Additionally, integrating diffusion-based trajectory generators with differentiable safety and collision-avoidance filters can produce smooth, compliant motions that align tightly with high-level task planning.

On the dataset side, procedural task grammars embedded in simulators can automatically generate long-horizon, open-ended scenarios that interleave navigation and fine-grained manipulation. To support sensor fusion, standardized multimodal capture pipelines should be adopted to

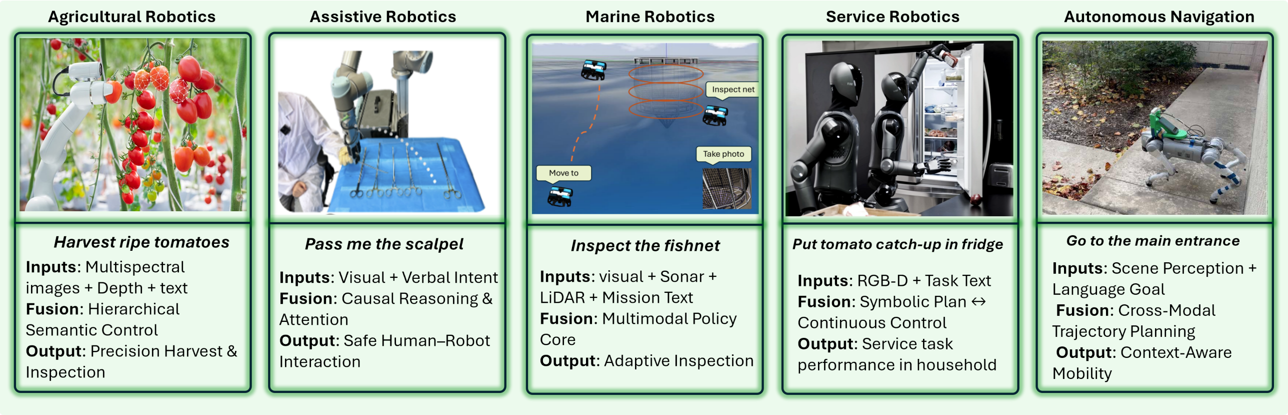

Agricultural Robotics
Harvest ripe tomatoes
Inputs: Multispectral images + Depth + text
Fusion: Hierarchical Semantic Control
Output: Precision Harvest & Inspection
Assistive Robotics
Pass me the scalpel
Inputs: Visual + Verbal Intent
Fusion: Causal Reasoning & Attention
Output: Safe Human–Robot Interaction
Marine Robotics
Inspect net
Take photo
Move to
Inspect the fishnet
Inputs: visual + Sonar + LiDAR + Mission Text
Fusion: Multimodal Policy Core
Output: Adaptive Inspection
Service Robotics
Put tomato catch-up in fridge
Inputs: RGB-D + Task Text
Fusion: Symbolic Plan ↔ Continuous Control
Output: Service task performance in household
Autonomous Navigation
Go to the main entrance
Inputs: Scene Perception + Language Goal
Fusion: Cross-Modal Trajectory Planning
Output: Context-Aware Mobility

Figure 21: Representative emerging application domains of VLA systems. Multimodal fusion enables grounded perception, reasoning, and control across: (a) *Agricultural Robotics*: precision harvesting and inspection; (b) *Assistive Robotics*: causal reasoning for safe human-robot interaction; (c) *Marine Robotics*: adaptive inspection under degraded sensing; (d) *Service Robotics*: symbolic-to-continuous task execution; and (e) *Autonomous Navigation*: context-aware mobility through cross-modal planning.

synchronize RGB-D, tactile, force/torque, audio, and language streams at compatible sampling rates, with missing modalities augmented through cross-modal synthesis (e.g., monocular depth estimation). Annotation burdens can be reduced through self-supervised or weakly supervised techniques, including unsupervised segmentation, vision-language co-training, and active learning, to automatically extract object masks, 6-DoF trajectories, and language explanations. Hybrid synthetic-real pipelines, using neural rendering and physics-aware domain randomization, can bridge the realism-scale gap, ensuring that large-scale simulated data generalize better to physical environments.

In simulation platforms, physics fidelity should be improved through differentiable, multi-scale contact models that blend classical solvers with data-driven calibration to better handle soft-body deformation, friction variability, and compliance. Hybrid rendering pipelines that combine high-throughput rasterization for general frames with neural or ray-traced rendering for key scenes can deliver realism without compromising speed. A simulator-agnostic language grounding API should be established to map natural language instructions directly to scene graphs and agent behaviors. Finally, to enable broad generalization, simulators must support multi-robot and multi-agent scenarios, with autoimport of URDF/SDF models and shared simulation protocols, allowing for consistent policy pretraining across heterogeneous robot platforms.

8.7.2. Emerging Application Domains

The growing maturity of VLA model frameworks marks a transition from controlled laboratory settings to real-world embodied systems capable of grounding abstract linguistic goals into sensorimotor execution. By unifying perception, reasoning, and control through hierarchical multimodal fusion, VLAs are balanced to redefine autonomy in several critical domains (as depicted in Fig. 21) where adaptability, safety, and explainability are critical.

- **Agricultural Robotics:** VLA-guided agricultural robots integrate multispectral vision, depth sensing, and environmental telemetry with language-conditioned task representations, enabling precise and context-aware decision-making. Commands such as *harvest the ripe tomatoes* or *inspect the southern greenhouse for disease* can be semantically linked to spatial features and phenotypic cues. Hierarchical fusion allows linguistic intent to influence perception modules, while lower layers extract visual affordances such as ripeness, occlusion, or leaf texture. Through multimodal grounding.

- **Assistive Robotics:** In medical and assistive contexts, VLA-driven systems provide interpretable interfaces between human intent and compliant robotic control. Natural-language instructions such as *pass me the scalpel* or *assist the patient to sit upright* are translated into safe motion primitives through causal reasoning and cross-modal attention. Information-theoretic quantities like entropy reduction (ΔH_k) quantify task confidence, while attention attribution ensures traceability between linguistic and visual grounding. These capabilities promote transparent, verifiable autonomy in clinical and home-care robotics, where safety and causal interpretability are essential.

- **Marine Robotics:** For surface and underwater platforms operating in GNSS-denied or visually degraded conditions, VLAs link semantic mission directives with fused perceptual streams (camera, sonar, LiDAR, and inertial data). Commands such as *inspect the fishnet* or *survey pen three for damage* are executed through multimodal reasoning that aligns linguistic intent with geometric and environmental context (Akram et al., 2025). Hierarchical fusion enhances robustness by maintaining cross-modal coherence under partial observability, while counterfactual analysis and attention monitoring identify failure points in perception-action coupling, improving mission safety and adaptability.

- **Service Robotics:** In manufacturing, logistics, and domestic environments, VLA policies enable high-level task programming through natural commands such as *put tomato catch-up in fridge* or *assemble the blue valve on rack three*. These instructions are decomposed through structured fusion layers into symbolic plans and continuous control policies, allowing flexible adaptation to workspace variation and operational uncertainty. Fusion-energy dynamics (E_{fusion}) and generalization indices (GI) serve as diagnostic indicators of robustness across visual, linguistic, and proprioceptive inputs, while attention transparency helps supervisors verify safety and quality compliance.
- **Autonomous Navigation:** For mobile, aerial, and humanoid platforms, VLAs facilitate context-aware navigation and interaction in dynamic environments. By coupling semantic scene understanding with natural-language intent such as *go to main entrance* or *monitor the corridor for visitors*, these systems infer spatial goals, predict social interactions, and plan motion trajectories using diffusion or flow-based policies. Hierarchical cross-modal fusion ensures that linguistic reasoning informs trajectory planning, while causal interpretability safeguards against ambiguous or unsafe responses. This synergy between multimodal grounding and policy transparency supports safe, explainable autonomy in open-world mobility applications.

Across these domains, VLAs act as a unifying paradigm that transforms high-level human intent into grounded, verifiable robotic behavior. The integration of causal interpretability, quantitative fusion metrics, and benchmark-driven evaluation not only advances performance but also establishes the theoretical and practical foundations for safe, transparent, and domain-adaptive embodied intelligence.

8.8. Agentic VLA Autonomy and Self-Improving Control

Building upon the theoretical and benchmarking foundations established in Sections 4-7, we now identify a crucial trajectory for next-generation multimodal autonomy: the emergence of Agentic VLA Robotics, where fusion models evolve from reactive policies into proactive, self-improving embodied agents Although current VLA pipelines combine perception, language grounding, and motor control, most of them still function as *reactive systems* that execute externally defined tasks without autonomous decision-making. An emerging direction for next-generation embodied intelligence is to extend VLA models into *agentic AI architectures* capable of formulating goals, reasoning about tasks, and autonomously selecting skills. In this emerging paradigm, VLA models would no longer serve as fixed end-policies, but instead act as reusable embodied tools that are called, verified, and improved by a higher-level cognitive agent.

Although the term *Agentic AI* is increasingly referenced in emerging robotics literature, its definition remains inconsistent across studies. To provide conceptual clarity and align terminology for embodied autonomy, we distinguish between three closely related but fundamentally different frameworks: *Agentic AI*, *Agentic VLMs*, and *Agentic VLAs*. These concepts represent a progression from high-level cognitive autonomy to grounded multimodal and visuomotor control.

- **Agentic AI** refers to artificial systems capable of *self-directed autonomy*, including goal generation, task decomposition, tool invocation, verification, feedback monitoring, and uncertainty-aware re-planning. Unlike classical agents that execute predefined policies, agentic systems incorporate deliberate cognitive functions such as reasoning, memory retrieval, introspective evaluation, and adaptive correction. Recent works (Raptis et al., 2025; Abou Ali et al., 2025; Park et al., 2025) demonstrate early agentic pipelines where in large language models autonomously trigger APIs, interpret sensory evidence, and refine task execution through iterative feedback loops
- **Agentic VLMs** extend this paradigm by integrating agentic reasoning with *vision–language grounding*. These models not only interpret scenes and instructions, but also perform agent-like operations such as verifying outcomes, detecting failures, and reasoning over semantic and spatial constraints. Studies such as (Liu et al., 2024a; Zhang et al., 2025f) highlight VLMs that operate as perceptual-verification modules in agentic controllers, enabling robots to update plans based on multimodal evidence and maintain semantic memory of task states.
- **Agentic VLAs** represent the most embodied form of agentic autonomy. In this setting, an LLM-based planner invokes VLA manipulation or navigation skills as callable tools, evaluates their outcomes through multimodal verification, and dynamically adapts execution. Recent examples (Abou Ali et al., 2025; Park et al., 2025) demonstrate pipelines where grounded visuomotor skills are embedded within an agentic loop capable of synthesizing subgoals, recovering from failures, and switching skills in a context-aware manner. Agentic VLAs therefore unify symbolic reasoning, multimodal perception, and action execution under a single closed feedback loop that enables robots to act proactively rather than reactively.

  These categories indicate a shift from static VLA pipelines toward adaptive, self-improving embodied systems.. The existing literature remains fragmented, lacking a unified framework that differentiates between agentic reasoning, multimodal grounding, and embodied control. By formalizing the distinctions between Agentic AI, Agentic VLMs, and Agentic VLAs, this review provides a clearer conceptual foundation for understanding these concepts. Places recent advances within a coherent roadmap for developing next-generation agent-driven robotic autonomy.

The closed-loop agentic pipeline that captures this transition is depicted in Fig. 22. A large language model functions as the *agentic planner*, breaking tasks into subgoals and selecting suitable perception or manipulation tools from a *Skill & Tool Library*. Rather than executing commands blindly, future systems may continuously monitor execution through a *VLM-based observation module* with memory buffers that store visual feedback. A dedicated verification step would then confirm whether a subgoal is achieved or requires re-planning. Importantly, progress could be assessed not only by task completion, but also by *feedback metrics* such as entropy reduction ΔH_k, fusion energy balance, and uncertainty measures. These metrics could guide replanning, tool switching, or adaptive refinement of physical actions. Such mechanisms would create a continuous loop in which planning, grounding, verification, and improvement occur continuously during execution.

Future agentic autonomy therefore has the potential to extend VLA capabilities beyond prompt-conditioned responses by enabling robots to: (i) generate and refine goals independently; (ii) select multimodal skills via API-based tool interfaces; (iii) verify actions through visual-language reasoning; and (iv) adapt their decisions based on multimodal confidence scores. Recent developments already point toward such architectures, where LLM-driven agents coordinate perception, motion strategies, and symbolic reasoning using feedback-driven API invocation (Raptis et al., 2025; Abou Ali et al., 2025; Park et al., 2025). These agentic systems are expected to learn *not only how to act*, but also *when to modify, repeat, or delegate actions*, leading to more transparent and explainable embodied decision processes.

From this perspective, we outline a potential **Agentic VLA Architecture** with two tightly connected future layers: (1) an LLM-driven reasoning layer responsible for self-goal generation, scheduling, verification, and failure recovery, and (2) a VLA skill layer offering reusable multimodal perception and manipulation tools. Real-time diagnostic signals, such as ΔH_k, E_{fusion}, and the VLA-FEB efficiency metric η_k (Sections 4.5-7.2), may serve as measurable criteria to advance decisions to humans, delegate tasks to other robots, or refine actions in real time.

Collaborative agentic autonomy is also a promising direction, where orchestration extends across multiple robotic platforms. Future LLM-based planners may assign roles, share multimodal states, and negotiate task procedures through natural language (Liu et al., 2024a; Zhang et al., 2025f). Embedding VLA tools into such frameworks can unlock multi-robot manipulation, shared perception under uncertainty, and truly open-world task execution.

We therefore view *Agentic VLA Robotics* as a future class of embodied systems in which VLA models operate as verifiable, self-improving tools governed by agentic controllers capable of generating goals, re-planning actions, interpreting uncertainty, and making collaborative decisions.

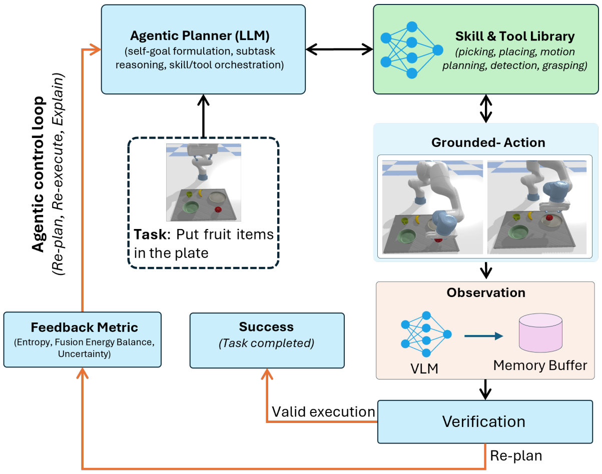

Agentic Planner (LLM)
(self-goal formulation, subtask reasoning, skill/tool orchestration)
Skill & Tool Library
(picking, placing, motion planning, detection, grasping)
Agentic control loop
(Re-plan, Re-execute, Explain)
Task: Put fruit items in the plate
Grounded- Action
Observation
VLM
Memory Buffer
Feedback Metric
(Entropy, Fusion Energy Balance, Uncertainty)
Success
(Task completed)
Valid execution
Verification
Re-plan

Figure 22: *Agentic VLA Robotics Framework.* An LLM-based agentic planner performs self-goal formulation and subtask scheduling, invoking VLA manipulation/perception tools from a skill library. A VLM-based observation module with memory provides visual grounding and verification. Real-time diagnostic feedback drives re-planning in the event of failure or advances execution upon success. This establishes a closed cognitive-embodied loop where reasoning, grounding, verification, and improvement are continuously coupled.

Advancing this paradigm will require: (i) safe interfaces between agent reasoning and embodied control; (ii) evaluation metrics such as VLA-FEB to measure long-horizon self-improvement and negotiation reliability; and (iii) scalable inference techniques that meet real-time constraints on physical robots. Together, agentic autonomy and VLA fusion have the potential to shift robotics from *reactive* policy execution toward explainable, adaptive, and self-directed embodied intelligence. Thus, agentic autonomy constitutes not only an architectural opportunity but also a prerequisite for scalable, trustworthy embodied intelligence capable of operating safely in unstructured environments.

8.9. Long-Horizon Autonomy and Agentic VLA Systems

While recent Agentic VLA frameworks demonstrate promising closed-loop behavior, several key challenges remain unresolved for long-horizon autonomy. First, *memory systems* are still under-developed. Most current pipelines rely on stateless or short-context reasoning, which limits their ability to handle multi-stage tasks, track evolving scene context, recall past failures, and maintain persistent semantic maps. Achieving reliable long-horizon behavior requires integration of episodic memory (task progress, failures, corrections) and semantic memory (object attributes, spatial layouts, affordances) tightly coupled with VLA skill libraries.

Symbolic reasoning integration is essential for complex, multi-step missions that require abstract planning beyond immediate perception. Although LLM-based planners can generate subgoals, their symbolic grounding remains fragile in cluttered or dynamic scenes. Future Agentic VLA architectures must combine symbolic structures, such as task graphs, precondition, effect models, and relational abstractions with grounded visuomotor execution, enabling more reliable plan consistency, hierarchical decomposition, and error recovery.

Cross-robot coordination remains largely unexplored. Heterogeneous multi-robot systems, such as UAV-UGV or manipulator-mobile base pairs, require synchronized reasoning about shared goals, spatial constraints, task allocation, and inter-robot communication. Current VLA models operate mainly on single-robot trajectories, lacking mechanisms for distributed memory, shared world models, or multi-agent policy alignment. Extending Agentic VLA loops to multi-robot settings will be crucial for scalable embodied intelligence. Bridging short-horizon agentic control with long-horizon autonomy will require integrating memory systems, symbolic structures, and coordinated multi-agent reasoning into a unified embodied intelligence framework. These dimensions represent critical frontiers for the next generation of Agentic VLA robotics.

9. Conclusion

This review synthesizes recent progress in VLA modelling by integrating architectural, dataset, and simulation perspectives into a unified analytical framework. The large-scale analysis reveals that hierarchical and late fusion architectures achieve the best balance between generalization and efficiency. Encoder scale and fusion depth are found to be key factors influencing manipulation success, while diffusion-based decoders consistently show superior robustness and cross-domain transfer compared to autoregressive variants. These results highlight the growing importance of balanced, multimodal fusion over simple parameter scaling in embodied learning. The analysis of available datasets reveals a persistent lack of benchmarks that combine complex task semantics with rich multimodal alignment, limiting the development of scalable and general-purpose robotic policies. Similarly, current simulation environments often lack synchronized multimodal acquisition and linguistic grounding, emphasizing the need for more comprehensive simulation-to-data pipelines and unified embodied evaluation frameworks. Overall, this study underscores the importance of enhancing data diversity and synchronization across vision, language, and proprioceptive modalities, developing adaptive fusion architectures capable of dynamically balancing modalities under uncertainty, and establishing transparent evaluation protocols that quantify alignment, interpretability, and transfer efficiency. Agentic VLA Robotics highlights a future shift toward embodied systems where VLA skills operate as callable, verifiable tools governed by LLM planners capable of real-time re-planning, uncertainty-aware adaptation, and collaborative autonomy. By integrating theoretical and empirical insights, this review offers a comprehensive reference and a forward-looking roadmap for the development of scalable, interpretable, and trustworthy embodied AI systems.

Declaration

Declaration of generative AI and AI-assisted technologies in the writing process. During the preparation of this work the author(s) used ChatGPT in order to refine the English and to search the names of the VLA models. After using this tool/service, the author(s) reviewed and edited the

References

Abdel-Hamid, A., Mahmoud, K., et al., 2024. Image captioning transformers: A comprehensive review. Artificial Intelligence Review doi:10.1007/s10462-024-10560-w. early access.

Abou Ali, M., Dornaika, F., Charafeddine, J., 2025. Agentic ai: a comprehensive survey of architectures, applications, and future directions. Artificial Intelligence Review 59. doi:10.1007/s10462-025-11422-4.

Ahn, M., Brohan, A., Brown, N., et al., 2022. Do as i can, not as i say: Grounding language in robotic affordances. arXiv preprint arXiv:2204.01691 .

Akram, W., Ud Din, M., Saad, A., Hussain, I., 2025. Aquachat: An llm-guided rov framework for adaptive inspection of aquaculture net pens. Aquaculture Engineering In press.

Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., van den Hengel, A., 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Arai, H., Miwa, K., Sasaki, K., Watanabe, K., Yamaguchi, Y., Aoki, S., Yamamoto, I., 2025. Covla: Comprehensive vision-language-action dataset for autonomous driving. 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), IEEE , 1933–1943.

Arai, H., Miwa, K., Sasaki, K., Yamaguchi, Y., Watanabe, K., Aoki, S., Yamamoto, I., 2024. Covla: Comprehensive vision-language-action dataset for autonomous driving URL: https://arxiv.org/abs/2408.10845, arXiv:2408.10845.

Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al., 2025. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. arXiv preprint arXiv:2506.09985 .

authors, A., 2025. Ef-vla: Vision-language-action models with aligned vision language features for better generalization. Under review at ICLR 2025 URL: https://openreview.net/forum?id=8512. preprint under double-blind review.

Bharadhwaj, H., Pore, N., Liang, J., Singh, J., Rao, K., Zeng, A., Gopalakrishnan, K., 2023. Roboagent: Generalist robot agent with semantic and temporal understanding. arXiv preprint arXiv:2310.08560 URL: https://robopen.github.io/media/roboagent.pdf.

Bjorck, J., Castañeda, F., Cherniadev, N., Da, X., Ding, R., Fan, L., Fang, Y., Fox, D., Hu, F., Huang, S., et al., 2025. Gr00t n1: An open foundation model for generalist humanoid robots. arXiv preprint arXiv:2503.14734 .

Black, K., Brown, N., Darpinian, J., Dhabalia, K., Driess, D., Esmail, A., Equi, M., et al., 2025. π-0.5:: A vision-language-action model with open-world generalization. arXiv preprint arXiv:2504.16054 URL: https://arxiv.org/abs/2504.16054.

Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al., 2024a. Pi-0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 .

Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al., 2024b. pi_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 .

Brandisauskas, M., Zukauskas, M., Krizhanovsky, A., 2023. Seq2code: Encoder-decoder model for program synthesis. Procedia Computer Science 222, 1441–1450. doi:10.1016/j.procs.2023.11.306.

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al., 2022. Rt-1: Robotics transformer for real-world control at scale. arXiv preprint arXiv:2212.06817 .

Brown, T.B., Mann, B., Ryder, N., et al., 2020. Language models are few-shot learners. Advances in Neural Information Processing Systems 33, 1877–1901.

Bu, Q., Cai, J., Chen, L., Cui, X., Ding, Y., Feng, S., Gao, S., He, X., Hu, X., Huang, X., et al., 2025. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. arXiv preprint arXiv:2503.06669 .

Budzianowski, P., Maa, W., Freed, M., Mo, J., Xie, A., Tipnis, V., Bolte, B., 2024. Edgevla: Efficient vision-language-action models. environments 20, 3.

Caron, M., Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H., 2021. Emerging properties in self-supervised vision transformers, in: ICCV.

Chen, P., Bu, P., Wang, Y., Wang, X., Wang, Z., Guo, J., Zhao, Y., Zhu, Q., Song, J., Yang, S., et al., 2025a. Combatvla: An efficient vision-language-action model for combat tasks in 3d action role-playing games. arXiv preprint arXiv:2503.09527 .

Chen, X., et al., 2024. Quar-vla: A vision-language-action model for quadruped robots. arXiv preprint arXiv:2310.08532 .

Chen, Y., Tian, S., Liu, S., Zhou, Y., Li, H., Zhao, D., 2025b. Conrft: A reinforced fine-tuning method for vla models via consistency policy. arXiv preprint arXiv:2502.05450 .

Chen, Z., Huo, J., Chen, Y., Gao, Y., 2025c. Robohorizon: An llm-assisted multi-view world model for long-horizon robotic manipulation. arXiv preprint arXiv:2501.06605 .

Cheng, A., Ji, Y., Yang, Z., Gongye, Z., Zou, X., Kautz, J., Bıyık, E., Yin, H., Liu, S., Wang, X., 2024. Navila: Legged robot vision-language-action model for navigation. arXiv preprint arXiv:2412.04453 .

Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S., 2023a. Diffusion policy: Visuomotor policy learning via action diffusion. The International Journal of Robotics Research 02783649241273668.

Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S., 2023b. Diffusion policy: Visuomotor policy learning via action diffusion. arXiv preprint arXiv:2303.04137 .

Chiang, H., Xu, Z., Fu, Z., Jacob, M., Zhang, T., Lee, T., Yu, W., Schenck, C., Rendleman, D., Shah, D., et al., 2024. Mobility vla: Multimodal instruction navigation with long-context vlms and topological graphs. arXiv preprint arXiv:2407.07775 .

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Sepassi, R., Gehrmann, S., Elsen, E., Patrick, D., Mishkin, P., 2022. Palm: Scaling language modeling with pathways. arXiv preprint arXiv:2204.02311 .

Collaboration, E., O'Neill, A., Rehman, A., Gupta, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., et al., A.P., 2025. Open x-embodiment: Robotic learning datasets and rt-x models. URL: https://arxiv.org/abs/2310.08864, arXiv:2310.08864.

Coumans, E., Bai, Y., 2016. Pybullet, a python module for physics simulation for robotics, games and machine learning. https://pybullet.org.

Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D., 2018. Embodied question answering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 248–255URL: https://doi.org/10.1109/CVPR.2009.5206848, doi:10.1109/CVPR.2009.5206848.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Dey, S., Zaech, J., Nikolov, N., Van Gool, L., Paudel, D., 2024. Revla: Reverting visual domain limitation of robotic foundation models. arXiv preprint arXiv:2409.15250 .

Din, M.U., Rosell, J., Akram, W., Zaplana, I., Roa, M.A., Hussain, I., 2025. Llm-guided task and motion planning using knowledge-based reasoning. URL: https://arxiv.org/abs/2412.07493, arXiv:2412.07493.

Ding, P., Ma, J., Tong, X., Zou, B., Luo, X., Fan, Y., Wang, T., Lu, H., Mo, P., Liu, J., et al., 2025. Humanoid-vla: Towards universal humanoid control with visual integration. arXiv preprint arXiv:2502.14795 .

Ding, P., Zhao, H., Zhang, W., Song, W., Zhang, M., Huang, S., Yang, N., Wang, D., 2024. Quar-vla: Vision-language-action model for quadruped robots. European Conference on Computer Vision, Springer , 352–367.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations (ICLR). URL: https://arxiv.org/abs/2010.11929.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

Driess, D., Ruiz, N., Goyal, K., Chebotar, Y., Irpan, A., Ailon, X., Levine, S., Finn, C., 2023. Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 .

Driess, D., Springenberg, J.T., Ichter, B., Yu, L., Li-Bell, A., Pertsch, K., Ren, A.Z., Walke, H., Vuong, Q., Shi, L.X., et al., 2025. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. arXiv preprint arXiv:2505.23705 .

Driess, D., et al., 2024. Openvla: Open-source vision-language-action models for robotics. arXiv preprint arXiv:2406.09246 .

Fan, C., Jia, X., Sun, Y., Wang, Y., Wei, J., Gong, Z., Zhao, X., Tomizuka, M., Yang, X., Yan, J., Ding, M., 2025. Interleave-vla: Enhancing robot manipulation with interleaved image-text instructions. arXiv preprint arXiv:2406.07000 URL: https://arxiv.org/abs/2406.07000.

Fang, H., Grotz, M., Pumacay, W., Wang, Y.R., Fox, D., Krishna, R., Duan, J., 2025. Sam2act: Integrating visual foundation model with a memory architecture for robotic manipulation. arXiv preprint arXiv:2501.18564 .

Fouad, S., et al., 2024. Image captioning using deep learning: A comprehensive review and future perspectives. Multimedia Tools and Applications doi:10.1007/s11042-024-17234-3.

Fu, H., Zhang, D., Zhao, Z., Cui, J., Liang, D., Zhang, C., Zhang, D., Xie, H., Wang, B., Bai, X., 2025. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. arXiv preprint arXiv:2503.19755 .

Gao, X., Gao, Q., Gong, R., Lin, K., Thattai, G., Sukhatme, G.S., 2022a. DialFRED: Dialogue-Enabled Agents for Embodied Instruction Following. IEEE Robotics and Automation Letters 7, 10049–10056. doi:10.1109/LRA.2022.3193254.

Gao, X., Gao, Q., Gong, R., Lin, K., Thattai, G., Sukhatme, G.S., 2022b. Dialfred: Dialogue-enabled agents for embodied instruction following. IEEE Robotics and Automation Letters 7, 10049–10056. URL: http://dx.doi.org/10.1109/LRA.2022.3193254, doi:10.1109/lra.2022.3193254.

Gbagbe, K.F., Cabrera, M.A., Alabbas, A., Alyunes, O., Lykov, A., Tsetserukou, D., 2024. Bi-vla: Vision-language-action model-based system for bimanual robotic dexterous manipulations. arXiv preprint arXiv:2405.06039 URL: https://arxiv.org/abs/2405.06039.

Ghojogh, B., Ghodsi, A., Karray, F., Crowley, M., 2024. Attention mechanism in machine learning: A survey. arXiv preprint arXiv:2402.05310 URL: https://arxiv.org/abs/2402.05310.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., et al., R.G., 2022. Ego4d: Around the world in 3,000 hours of egocentric video URL: https://arxiv.org/abs/2110.07058, arXiv:2110.07058.

Gu, J., Kirmani, S., Wohlhart, P., Lu, Y., Arenas, M.G., Rao, K., Yu, W., Fu, C., Gopalakrishnan, K., Xu, Z., et al., 2023. Robotic task generalization via hindsight trajectory sketches. arXiv preprint arXiv:2311.01977 URL: https://arxiv.org/abs/2311.01977.

Guo, Y., Zhang, J., Chen, X., Ji, X., Wang, Y.J., Hu, Y., Chen, J., 2025. ire-vla: Improving vision-language-action model with online reinforcement learning. arXiv preprint arXiv:2501.16664 URL: https://arxiv.org/abs/2501.16664.

Guruprasad, P., Sikka, H., Song, J., Wang, Y., Liang, P.P., 2024. Benchmarking vision, language, & action models on robotic learning tasks. arXiv preprint arXiv:2411.05821 .

Han, S., Qiu, B., Liao, Y., Huang, S., Gao, C., Yan, S., Liu, S., 2025. Robocerebra: A large-scale benchmark for long-horizon robotic manipulation evaluation. arXiv preprint arXiv:2506.06677 .

Hao, P., Zhang, C., Li, D., Cao, X., Hao, X., Cui, S., Wang, S., 2025. Tla: Tactile-language-action model for contact-rich manipulation. arXiv preprint arXiv:2503.08548 URL: https://arxiv.org/abs/2503.08548.

He, Y., Weilbach, C.D., Wojciechowska, M.E., Zhang, Y., Wood, F., 2025. Plaicraft: Large-scale time-aligned vision-speech-action dataset for embodied ai. URL: https://arxiv.org/abs/2505.12707, arXiv:2505.12707.

Hou, Z., Zhang, T., Xiong, Y., Pu, H., Zhao, C., Tong, R., Qiao, Y., Dai, J., Chen, Y., 2024a. Diffusion transformer policy. arXiv preprint arXiv:2410.15959 .

Hou, Z., Zhang, T., Xiong, Y., Pu, H., Zhao, C., Tong, R., Qiao, Y., Dai, J., Chen, Y., 2024b. Diffusion transformer policy: Scaling diffusion transformer for generalist vision–language–action learning. arXiv preprint arXiv:2410.15959 .

Hu, J., Hendrix, R., Farhadi, A., Kembhavi, A., Martín-Martín, R., Stone, P., Zeng, K.H., Ehsani, K., 2024. Flare: Achieving masterful and adaptive robot policies with large-scale reinforcement learning fine-tuning. arXiv preprint arXiv:2409.16578 .

Huang, H., Liu, F., Fu, L., Wu, T., Mukadam, M., Malik, J., Goldberg, K., Abbeel, P., 2025a. Otter: A vision-language-action model with text-aware visual feature extraction. arXiv preprint arXiv:2503.03734 URL: https://arxiv.org/abs/2503.03734.

Huang, S., Chang, H., Liu, Y., Zhu, Y., Dong, H., Boularias, A., Gao, P., Li, H., 2024. A3vlm: Actionable articulation-aware vision language model, in: Proceedings of the 8th Conference on Robot Learning (CoRL). URL: https://arxiv.org/abs/2405.06039.

Huang, S., Chen, L., Zhou, P., Chen, S., Jiang, Z., Hu, Y., Liao, Y., Gao, P., Li, H., Yao, M., et al., 2025b. Enerverse: Envisioning embodied future space for robotics manipulation. arXiv preprint arXiv:2501.01895 .

Huang, W., Wang, C., Zhang, R., Li, Y., Wu, J., Fei-Fei, L., 2023. Voxposer: Composable 3d value maps for robotic manipulation with language models. arXiv preprint arXiv:2307.05973 .

Hung, C., Sun, Q., Hong, P., Zadeh, A., Li, C., Tan, U., Majumder, N., Poria, S., et al., 2025. Nora: A small open-sourced generalist vision language action model for embodied tasks. arXiv preprint arXiv:2504.19854 .

Jaegle, A., Gimeno, N., Brock, A., Zisserman, A., Carreira, J., Vinyals, O., Verdegaal, R., Pessoa, P., Nowozin, S., 2022. Perceiver IO: A general architecture for structured inputs & outputs, in: International Conference on Learning Representations (ICLR).

James, S., Ma, Z., Arrojo, D.R., Davison, A.J., 2020. Rlbench: The robot learning benchmark & learning environment. IEEE Robotics and Automation Letters 5, 3019–3026. doi:10.1109/LRA.2020.2972831.

Ji, Y., Tan, H., Shi, J., Hao, X., Zhang, Y., Zhang, H., Wang, P., Zhao, M., Mu, Y., An, P., et al., 2025. Robobrain: A unified brain model for robotic manipulation from abstract to concrete, in: Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 1724–1734.

Jiang, S., Li, H., Ren, R., Zhou, Y., Wang, Z., He, B., 2025. Kaiwu: A multimodal manipulation dataset and framework for robot learning and human-robot interaction. URL: https://arxiv.org/abs/2503.05231, arXiv:2503.05231.

Jiang, Y., Gupta, A., Zhang, Z., et al., 2022. Vima: General robot manipulation with multimodal prompts. arXiv preprint arXiv:2210.03094 .

Jones, J., Mees, O., Sferrazza, C., Stachowicz, K., Abbeel, P., Levine, S., 2025. Beyond sight: Finetuning generalist robot policies with heterogeneous sensors via language grounding. arXiv preprint arXiv:2501.04693 .

Juliani, A., Berges, V., Teng, E., Gao, Y., Henry, H., Mattar, M., Lange, D., 2018. Unity: A general platform for intelligent agents, in: Proceedings of the 1st Annual Conference on Robot Learning (CoRL), pp. 49–60.

Kang, G.C., Kim, J., Shim, K., Lee, J.K., Zhang, B.T., 2024. Clip-rt: Learning language-conditioned robotic policies from natural language supervision. arXiv preprint arXiv:2411.00508 .

Kang, G.C., Kim, J., Shim, K., Lee, J.K., Zhang, B.T., 2025. Clip-rt: Learning language-conditioned robotic policies from natural language supervision, in: Proceedings of Robotics: Science and Systems (RSS).

Kawaharazuka, K., Oh, J., Yamada, J., Posner, I., Zhu, Y., 2025. Vision–language–action models for robotics: A review towards real-world applications. IEEE Access 13, 162467–162504. URL: https://ieeexplore.ieee.org/abstract/document/11164279, doi:10.1109/ACCESS.2025.3609980.

Khan, M., Asfaw, S., Iarchuk, D., Cabrera, M., Moreno, L., Tokmurziyev, I., Tsetserukou, D., 2025. Shake-vla: Vision-language-action model-based system for bimanual robotic manipulations and liquid mixing. arXiv preprint arXiv:2501.06919 .

Khazatsky, A.e.a., 2024. Droid: A large-scale in-the-wild robot manipulation dataset, in: Robotics: Science and Systems (RSS). URL: https://droid-dataset.github.io/.

Kim, M., Finn, C., Liang, P., 2025. Fine-tuning vision-language-action models: Optimizing speed and success. arXiv preprint arXiv:2502.19645 .

Kim, M., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al., 2024. Openvla: An open-source vision-language-action model. arXiv preprint arXiv:2406.09246 .

Koenig, N., Howard, A., 2004. Design and use paradigms for gazebo, an open-source multi-robot simulator, in: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 2149–2154.

Kolve, E., Mottaghi, R., Han, W., Randhavane, T., Zheng, X., Li, Y., Gupta, A., Farhadi, A., 2017. AI2-THOR: An interactive 3d environment for visual ai, in: Proceedings of the 1st Annual Conference on Robot Learning (CoRL). URL: https://ai2thor.allenai.org.

Krantz, J., Wijmans, E., Mukhopadhyay, A., Lee, S., Chernova, S., Batra, D., 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments, in: Proceedings of the European Conference on Computer Vision (ECCV), Springer. pp. 104–120. URL: https://arxiv.org/abs/2004.07787, doi:10.1007/978-3-030-58568-6_7.

Lam, C., Wang, X., Lu, X., Yao, Y., Yang, M.H., 2023. Deep learning with vision transformers: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 13101–13124. doi:10.1109/TPAMI.2023.3241477.

Li, D., Peng, B., Li, C., Qiao, N., Zheng, Q., Sun, L., Qin, Y., Li, B., Luan, Y., Wu, B., et al., 2025a. An atomic skill library construction method for data-efficient embodied manipulation. arXiv preprint arXiv:2501.15068 .

Li, J., Li, X., Li, X., Huang, J., Zhang, J., Wang, L., Dou, Q., Ling, H., 2022. Align before fuse: Vision and language representation learning with momentum distillation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18713–18723.

Li, J., Peng, E.A., Wang, C., Liu, J., Feichtenhofer, C., 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10965–10975.

Li, J., Zhu, Y., Tang, Z., Wen, J., Zhu, M., Liu, X., Li, C., Cheng, R., Peng, Y., Feng, F., 2024a. Improving vision-language-action models via chain-of-affordance. arXiv preprint arXiv:2412.20451 .

Li, M., Wang, Z., He, K., Ma, X., Liang, Y., 2025b. Jarvis-vla: Post-training large-scale vision language models to play visual games with keyboards and mouse. arXiv preprint arXiv:2503.16365 .

Li, Q., Liang, Y., Wang, Z., Luo, L., Chen, X., Liao, M., Wei, F., Deng, Y., Xu, S., Zhang, Y., et al., 2024b. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. arXiv preprint arXiv:2411.19650 .

Li, S., Wang, J., Dai, R., Ma, W., Ng, W., Hu, Y., Li, Z., 2024c. Robonurse-vla: Robotic scrub nurse system based on vision-language-action model. arXiv preprint arXiv:2409.19590 .

Li, Y., Deng, Y., Zhang, J., Jang, J., Memmel, M., Yu, R., Garrett, C.R., Ramos, F., Fox, D., Li, A., et al., 2025c. Hamster: Hierarchical action models for open-world robot manipulation. arXiv preprint arXiv:2502.05485 .

Li, Y., Yan, G., Macaluso, A., Ji, M., Zou, X., Wang, X., 2025d. Integrating lmm planners and 3d skill policies for generalizable manipulation. arXiv preprint arXiv:2501.18733 .

Liang, L., Bian, L., Xiao, C., et al., 2023. Robo360: A 3d omnispective multi-material robotic manipulation dataset. arXiv preprint arXiv:2312.06686 .

Lin, F., Nai, R., Hu, Y., You, J., Zhao, J., Gao, Y., 2025. Onetwovla: A unified vision-language-action model with adaptive reasoning. arXiv preprint arXiv:2505.11917 URL: https://arxiv.org/abs/2505.11917.

Lin, K., Li, L., Gao, D., Yang, Z., Wu, S., Bai, Z., Lei, W., Wang, L., Shou, M., 2024. Showui: One vision-language-action model for gui visual agent. arXiv preprint arXiv:2411.17465 .

Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., Stone, P., 2023. Libero: Benchmarking knowledge transfer for lifelong robot learning. Advances in Neural Information Processing Systems 36, 44776–44791.

Liu, H., Zhang, S., Wang, J., 2024a. Large language models for multi-agent coordination: A survey. URL: https://arxiv.org/abs/2409.00011, arXiv:2409.00011.

Liu, J., Chen, H., An, P., Liu, Z., Zhang, R., Gu, C., Li, X., Guo, Z., Chen, S., Liu, M., et al., 2025. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. arXiv preprint arXiv:2503.10631 .

Liu, J., Liu, M., Wang, Z., An, P., Li, X., Zhou, K., Yang, S., Zhang, R., Guo, Y., Zhang, S., 2024b. Robomamba: Efficient vision-language-action model for robotic reasoning and manipulation. Advances in Neural Information Processing Systems 37, 40085–40110.

Liu, S., Wu, L., Li, B., Tan, H., Chen, H., Wang, Z., Xu, K., Su, H., Zhu, J., 2024c. Rdt-1b: a diffusion foundation model for bimanual manipulation. arXiv preprint arXiv:2410.07864 .

Liu, X., Chen, W., Chen, Y., Chen, Y.S., Wang, W.Y., 2021. Pre-train or prompt? exploring the encoder-decoder framework for zero-shot learning. arXiv preprint arXiv:2104.08691 URL: https://arxiv.org/abs/2104.08691.

Lykov, A., Serpiva, V., Khan, M.H., Sautenkov, O., Myshlyaev, A., Tadevosyan, G., Yaqoot, Y., Tsetserukou, D., 2025. Cognitivedrone: A vla model and evaluation benchmark for real-time cognitive task solving and reasoning in uavs. arXiv preprint arXiv:2503.01378 .

Makoviychuk, V., Wawrzyniak, L., Rathod, Y., Allshire, A., Handa, A., Müller, J., Widmaier, F., Leal-Taixé, L., Makadia, A., Leutenegger, S., 2021. Isaac gym: High performance gpu based physics simulation for robot learning. arXiv preprint arXiv:2108.10470 URL: https://arxiv.org/abs/2108.10470.

Mees, O., Hermann, L., Rosete-Beas, E., Burgard, W., 2022. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. URL: https://arxiv.org/abs/2112.03227, arXiv:2112.03227.

Michel, O., 2004. Webots: Professional mobile robot simulation, in: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 4020–4025.

Mishra, A., Mishra, A., Singh, G., et al., 2024. Image transformers: A survey. ACM Computing Surveys In press.

Myers, V., Zheng, B.C., Dragan, A., Fang, K., Levine, S., 2025. Temporal representation alignment: Successor features enable emergent compositionality in robot instruction following temporal representation alignment. arXiv preprint arXiv:2502.05454 .

Niu, D., Sharma, Y., Xue, H., Biamby, G., Zhang, J., Ji, Z., Darrell, T., Herzig, R., 2025. Pre-training auto-regressive robotic models with 4d representations. arXiv preprint arXiv:2502.13142 .

NVIDIA Corporation, . NVIDIA Isaac Sim. https://developer.nvidia.com/isaac-sim. Accessed: 2025-05-18.

Padmakumar, A., Thomason, J., Shrivastava, A., Lange, P., Narayan-Chen, A., Gella, S., Piramuthu, R., Tur, G., Hakkani-Tur, D., 2021. Teach: Task-driven embodied agents that chat. URL: https://arxiv.org/abs/2110.00534, arXiv:2110.00534.

Parada, C., Team, G.R., 2025. Gemini robotics on-device brings ai to local robotic devices. DeepMind Blog. Available at: https://deepmind.google/discover/blog/gemini-robotics-on-device-brings-ai-to-local-robotic-devices/.

Park, T.H., Choi, Y.J., Shin, S.H., Lee, C.E., Lee, K., 2025. La-rcs: Llm-agent-based robot control system. Sensors and Materials 37, 3073–3089. URL: https://sensors.myu-group.co.jp/sm_pdf/SM4104.pdf, doi:10.18494/SAM5643.

Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., Levine, S., 2025. Fast: Efficient action tokenization for vision-language-action models. arXiv preprint arXiv:2501.09747 .

Pfeiffer, J., Vulic, I., Gurevych, I., 2020. Adapterfusion: Non-destructive task composition for transfer learning, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 4875–4884.

Press, O., Wolf, L., 2017. Using the output embedding to improve language models, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL), pp. 157–163. URL: https://arxiv.org/abs/1608.05859.

Qi, Z., Zhang, W., Ding, Y., Dong, R., Yu, X., Li, J., Xu, L., Li, B., He, X., Fan, G., et al., 2025. Sofar: Language-grounded orientation bridges spatial reasoning and object manipulation. arXiv preprint arXiv:2502.13143 .

Qu, D., Song, H., Chen, Q., Yao, Y., Ye, X., Ding, Y., Wang, Z., Gu, J., Zhao, B., Wang, D., et al., 2025. Spatialvla: Exploring spatial representations for visual-language-action model. arXiv preprint arXiv:2501.15830 .

Radford, A., Kim, J.W., Hallacy, C., et al., 2021. Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020 .

Raptis, E.K., Kapoutsis, A.C., Kosmatopoulos, E.B., 2025. Agentic llm-based robotic systems for real-world applications: a review on their agenticness and ethics. Frontiers in Robotics and AI 12. URL: https://link.springer.com/article/10.1007/s10462-025-11422-4, doi:10.3389/frobt.2025.1605405.

Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J., et al., 2022. A generalist agent. arXiv preprint arXiv:2205.06175 .

Rohmer, E., Singh, S.P., Freese, M., 2013. V-rep: A versatile and scalable robot simulation framework, in: 2013 IEEE/RSJ international conference on intelligent robots and systems, IEEE. pp. 1321–1326.

Samson, M., Muraccioli, B., Kanehiro, F., 2024. Scalable, training-free visual language robotics: A modular multi-model framework for consumer-grade gpus. arXiv preprint arXiv:2502.01071 URL: https://arxiv.org/abs/2502.01071.

Sapkota, R., Cao, Y., Roumeliotis, K.I., Karkee, M., 2025. Vision–language–action models: Concepts, progress, applications and challenges abs/2505.04769. URL: https://arxiv.org/abs/2505.04769, doi:10.48550/arXiv.2505.04769.

Sautenkov, O., Yaqoot, Y., Lykov, A., Mustafa, M., Tadevosyan, G., Akhmetkazy, A., Cabrera, M., Martynov, M., Karaf, S., Tsetserukou, D., 2025. Uav-vla: Vision-language-action system for large scale aerial mission generation. arXiv preprint arXiv:2501.05014 .

Savva, M., Chang, A.X., Dosovitskiy, A., et al., 2019. Habitat: A platform for embodied ai research. Proceedings of the IEEE/CVF International Conference on Computer Vision , 9339–9347.

Shi, L.X., Ichter, B., Equi, M., Ke, L., Pertsch, K., Vuong, Q., Tanner, J., Walling, A., Wang, H., Fusai, N., et al., 2025. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. arXiv preprint arXiv:2502.19417 .

Shridhar, M., Manuelli, L., Fox, D., 2022a. Cliport: What and where pathways for robotic manipulation. Conference on robot learning, PMLR , 894–906.

Shridhar, M., Manuelli, L., Fox, D., 2022b. Perceiver-actor: A multi-task transformer for robotic manipulation, in: Conference on Robot Learning (CoRL). URL: https://peract.github.io/paper/peract_corl2022.pdf.

Shridhar, M., Manuelli, L., Fox, D., 2023. Perceiver-actor: A multi-task transformer for robotic manipulation, in: Conference on Robot Learning, PMLR. pp. 785–799.

Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., Fox, D., 2020. ALFRED: a benchmark for interpreting grounded instructions for everyday tasks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). URL: https://arxiv.org/abs/1912.01734.

Shukor, M., Aubakirova, D., Capuano, F., Kooijmans, P., Palma, S., Zouitine, A., et al., 2025. Smolvla: A vision-language-action model for affordable and efficient robotics. arXiv preprint arXiv:2506.01844 URL: https://arxiv.org/abs/2506.01844.

Sliwowski, D., Jadav, S., Stanovcic, S., Orbik, J., Heidersberger, J., Lee, D., 2025. Reassemble: A multimodal dataset for contact-rich robotic assembly and disassembly. arXiv preprint arXiv:2502.05086 .

Song, C.H., Blukis, V., Tremblay, J., Tyree, S., Su, Y., Birchfield, S., 2025a. Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. URL: https://arxiv.org/abs/2411.16537, arXiv:2411.16537.

Song, W., Chen, J., Ding, P., Zhao, H., Zhao, W., Zhong, Z., Ge, Z., Ma, J., Li, H., 2025b. Accelerating vision-language-action model integrated with action chunking via parallel decoding. arXiv preprint arXiv:2503.02310 .

Tang, W., Pan, J.H., Liu, Y.H., Tomizuka, M., Li, L.E., Fu, C.W., Ding, M., 2025. Geomanip: Geometric constraints as general interfaces for robot manipulation. arXiv preprint arXiv:2501.09783 .

Team, F.A., 2025a. Helix: A vision-language-action model for generalist humanoid control. Available at: https://www.figure.ai/news/helix.

Team, G.R., 2025b. Gemini robotics: Bringing ai into the physical world. arXiv preprint arXiv:2503.20020 URL: https://arxiv.org/abs/2503.20020.

Team, O.M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., et al., 2024. Octo: An open-source generalist robot policy. arXiv preprint arXiv:2405.12213 .

Thomason, J., Murray, M., Cakmak, M., Zettlemoyer, L., 2019. Vision-and-dialog navigation. URL: https://arxiv.org/abs/1907.04957, arXiv:1907.04957.

Todorov, E., Erez, T., Tassa, Y., 2012. Mujoco: A physics engine for model-based control, in: 2012 IEEE/RSJ international conference on intelligent robots and systems, IEEE. pp. 5026–5033.

Touvron, H., Martin, T., Stone, L., Albert, A., Almahairi, A., Laradji, I., Aqaj, Y., Baratin, A., Lee, S., Verde, Z., Kaplanyan, A., Azar, M., Gelly, S., Joulin, A., 2023. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 .

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need, in: Advances in Neural Information Processing Systems (NeurIPS), pp. 5998–6008. URL: https://arxiv.org/abs/1706.03762.

Walke, H.R., Black, K., Zhao, T.Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A.W., Myers, V., Kim, M.J., Du, M., et al., 2023. Bridgedata v2: A dataset for robot learning at scale, in: Conference on Robot Learning, PMLR. pp. 1723–1736.

Wang, L., Zhang, H., Zhao, Y., Liu, Z., Bian, J., Yu, H., Xu, C., Lau, R., Wang, S., 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts, in: ACM International Conference on Multimedia (MM).

Wang, R., Zhang, J., Chen, J., Xu, Y., Li, P., Liu, T., Wang, H., 2023. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation, in: 2023 IEEE International Conference on Robotics and Automation (ICRA), IEEE. pp. 11359–11366.

Wang, S., Shan, J., Zhang, J., Gao, H., Han, H., Chen, Y., Wei, K., Zhang, C., Wong, K., Zhao, J., et al., 2025. Robobert: An end-to-end multimodal robotic manipulation model. arXiv preprint arXiv:2502.07837 .

Wang, T., Han, C., Liang, J.C., Yang, W., Liu, D., Zhang, L.X., Wang, Q., Luo, J., Tang, R., 2024a. Exploring the adversarial vulnerabilities of vision-language-action models in robotics. arXiv preprint arXiv:2411.13587 .

Wang, Z., Zheng, H., Nie, Y., Xu, W., Wang, Q., Ye, H., Li, Z., Zhang, K., Cheng, X., Dong, W., Cai, C., Lin, L., Zheng, F., Liang, X., 2024b. All robots in one: A new standard and unified dataset for versatile, general-purpose embodied agents. arXiv preprint arXiv:2408.10899 URL: https://arxiv.org/abs/2408.10899.

Wang, Z., Zhou, Z., Song, J., Huang, Y., Shu, Z., Ma, L., 2024c. Towards testing and evaluating vision-language-action models for robotic manipulation: An empirical study. arXiv preprint arXiv:2409.12894 .

Wang, Z., Zhou, Z., et al., 2024d. Ladev: A language-driven testing and evaluation platform for vision-language-action models in robotic manipulation. arXiv preprint arXiv:2410.05191 .

Wen, J., Zhu, M., Zhu, Y., Tang, Z., Li, J., Zhou, Z., Li, C., Liu, X., Peng, Y., Shen, C., Feng, F., 2024. Diffusion-vla: Generalizable and interpretable robot foundation model via self-generated reasoning. arXiv preprint arXiv:2412.03293 Accepted by ICML 2025.

Wen, J., Zhu, Y., Li, J., Tang, Z., Shen, C., Feng, F., 2025. Dexvla: Vision-language model with plug-in diffusion expert for general robot control. arXiv preprint arXiv:2502.05855 .

Wu, Y., Tian, R., Swamy, G., Bajcsy, A., 2025. From foresight to forethought: Vlm-in-the-loop policy steering via latent alignment. arXiv preprint arXiv:2502.01828 .

Xia, F., Li, C., Martín-Martín, R., Litany, O., Zamir, A.R., Savarese, S., 2020. Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5891–5898. URL: https://arxiv.org/abs/2002.10322, doi:10.1109/IROS45743.2020.9341201.

Xiang, F., Qin, Y., Mo, K., Xia, Y., Zhu, H., Liu, F., Liu, M., Jiang, H., Yuan, Y., Wang, H., et al., 2020. Sapien: A simulated part-based interactive environment, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11097–11107.

Xiao, B., Wu, H., Xu, W., Dai, X., Hu, H., Lu, Y., Zeng, M., Liu, C., Yuan, L., 2024. Florence-2: Advancing a unified representation for a variety of vision tasks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4818–4829.

Xu, B., Ud Din, M., Hussain, I., 2025a. Conditional variational auto encoder based dynamic motion for multitask imitation learning. Scientific Reports 15, 9196. URL: https://doi.org/10.1038/s41598-025-93888-4, doi:10.1038/s41598-025-93888-4.

Xu, S., Wang, Y., Xia, C., Zhu, D., Huang, T., Xu, C., 2025b. Vla-cache: Towards efficient vision-language-action model via adaptive token caching in robotic manipulation. arXiv preprint arXiv:2502.02175 .

Xue, H., Huang, X., Niu, D., Liao, Q., Kragerud, T., Gravdahl, J.T., Peng, X.B., Shi, G., Darrell, T., Sreenath, K., Sastry, S., 2025. Leverb: Humanoid whole-body control with latent vision-language instruction. arXiv preprint arXiv:2506.13751 URL: https://arxiv.org/abs/2506.13751.

Yan, F., Liu, F., Zheng, L., Zhong, Y., Huang, Y., Guan, Z., Feng, C., Ma, L., 2024. Robomm: All-in-one multimodal large model for robotic manipulation. arXiv preprint arXiv:2412.07215 URL: https://arxiv.org/abs/2412.07215.

Yang, J., Tan, R., Wu, Q., Zheng, R., Peng, B., Liang, Y., Gu, Y., Cai, M., Ye, S., Jang, J., et al., 2025. Magma: A foundation model for multimodal ai agents, in: Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 14203–14214.

Yang, M., Du, Y., Ghasemipour, K., Tompson, J., Schuurmans, D., Abbeel, P., 2023. Learning interactive real-world simulators. arXiv preprint arXiv:2310.06114 1, 6.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., Levine, S., 2020. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning, in: Conference on robot learning, PMLR. pp. 1094–1100.

Zawalski, M., Chen, W., Pertsch, K., Mees, O., Finn, C., Levine, S., 2025. Robotic control via embodied chain-of-thought reasoning. arXiv preprint arXiv:2407.08693 URL: https://arxiv.org/abs/2407.08693.

Zayyanu, M., Usman, B.T., Muda, Z., Salim, N.A., Mohamed, A., Abubakar, A.Y., Al-Obeidat, F., Malik, M.A., 2024. Revolutionising natural language processing with transformers: A survey. Information Processing & Management 61, 103528. doi:10.1016/j.ipm.2023.103528.

Zhang, B., Zhang, Y., Ji, J., Lei, Y., Dai, J., Chen, Y., Yang, Y., 2025a. Safevla: Towards safety alignment of vision-language-action model via safe reinforcement learning. arXiv e-prints , arXiv–2503.

Zhang, H., Ding, P., Lyu, S., Peng, Y., Wang, D., 2025b. Gevrm: Goal-expressive video generation model for robust visual manipulation. arXiv preprint arXiv:2502.09268 .

Zhang, H., Zantout, N., Kachana, P., Zhang, J., Wang, W., 2025c. Iref-vla: A benchmark for interactive referential grounding with imperfect language in 3d scenes. arXiv preprint arXiv:2503.17406 .

Zhang, J., Guo, Y., Chen, X., Wang, Y.J., Hu, Y., Shi, C., Chen, J., 2024a. Hirt: Enhancing robotic control with hierarchical robot transformers, in: Proceedings of the 8th Conference on Robot Learning (CoRL). URL: https://arxiv.org/abs/2410.05273.

Zhang, J., Guo, Y., Hu, Y., Chen, X., Zhu, X., Chen, J., 2025d. Up-vla: A unified understanding and prediction model for embodied agent. arXiv preprint arXiv:2501.18867 .

Zhang, J., Wang, K., Wang, S., Li, M., Liu, H., Wei, S., Wang, Z., Zhang, Z., Wang, H., 2024b. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. arXiv preprint arXiv:2412.06224 .

Zhang, R., Dong, M., Zhang, Y., Heng, L., Chi, X., Dai, G., Du, L., Wang, D., Du, Y., Zhang, S., 2025e. Mole-vla: Dynamic layer-skipping vision language action model via mixture-of-layers for efficient robot manipulation. arXiv preprint arXiv:2503.20384 .

Zhang, Y., Chen, R., Zhou, W., 2025f. Collaborative embodied agents via language-based negotiation and shared perception. International Journal of Robotics Research URL: https://journals.sagepub.com/. in Press.

Zhang, Z., Zheng, K., Chen, Z., Jang, J., Li, Y., Han, S., Wang, C., Ding, M., Fox, D., Yao, H., 2024c. Grape: Generalizing robot policy via preference alignment. arXiv preprint arXiv:2411.19309 .

Zhao, H., Song, W., Wang, D., Tong, X., Ding, P., Cheng, X., Ge, Z., 2025a. More: Unlocking scalability in reinforcement learning for quadruped vision-language-action models. arXiv preprint arXiv:2503.08007 .

Zhao, T., Kumar, V., Levine, S., Finn, C., 2023. Learning fine-grained bimanual manipulation with low-cost hardware. arXiv preprint arXiv:2304.13705 .

Zhao, W., Ding, P., Zhang, M., Gong, Z., Bai, S., Zhao, H., Wang, D., 2025b. Vlas: Vision-language-action model with speech instructions for customized robot manipulation. arXiv preprint arXiv:2502.13508 .

Zhao, W., Li, G., Gong, Z., Ding, P., Zhao, H., Wang, D., 2025c. Unveiling the potential of vision-language-action models with open-ended multimodal instructions. arXiv preprint arXiv:2505.11214 URL: https://arxiv.org/abs/2505.11214.

Zhen, H., Qiu, X., Chen, P., Yang, J., Yan, X., Du, Y., Hong, Y., Gan, C., 2024. 3d-vla: A 3d vision-language-action generative world model. arXiv preprint arXiv:2403.09631 URL: https://arxiv.org/abs/2403.09631.

Zheng, J., Li, J., Liu, D., Zheng, Y., Wang, Z., Ou, Z., Liu, Y., Liu, J., Zhang, Y.Q., Zhan, X., 2025a. Universal actions for enhanced embodied foundation models. arXiv preprint arXiv:2501.10105 .

Zheng, R., Liang, Y., Huang, S., Gao, J., III, H.D., Kolobov, A., Huang, F., Yang, J., 2025b. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies, in: International Conference on Learning Representations (ICLR). URL: https://arxiv.org/abs/2412.10345.

Zhong, Y., Bai, F., Cai, S., Huang, X., Chen, Z., Zhang, X., Wang, Y., Guo, S., Guan, T., Lui, K.N., Qi, Z., Liang, Y., Chen, Y., Yang, Y., 2025a. A survey on vision–language–action models: An action tokenization perspective abs/2507.01925. URL: https://arxiv.org/abs/2507.01925, doi:10.48550/arXiv.2507.01925.

Zhong, Y., Huang, X., Li, R., Zhang, C., Liang, Y., Yang, Y., Chen, Y., 2025b. Dexgraspvla: A vision-language-action framework towards general dexterous grasping. arXiv preprint arXiv:2502.20900 .

Zhou, E., An, J., Chi, C., Han, Y., Rong, S., Zhang, C., Wang, P., Wang, Z., Huang, T., Sheng, L., Zhang, S., 2025a. Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. URL: https://arxiv.org/abs/2506.04308, arXiv:2506.04308.

Zhou, Z., Zhu, Y., Zhu, M., Wen, J., Liu, N., Xu, Z., Meng, W., Cheng, R., Peng, Y., Shen, C., et al., 2025b. Chatvla: Unified multimodal understanding and robot control with vision-language-action model. arXiv preprint arXiv:2502.14420 .

Zhu, M., Zhu, Y., Li, J., Zhou, Z., Wen, J., Liu, X., Shen, C., Peng, Y., Feng, F., 2025a. Objectvla: End-to-end open-world object manipulation without demonstration. arXiv preprint arXiv:2502.19250 .

Zhu, T., Zhang, H., Zhou, Y., Yu, W., Wang, Y., Ma, L., Xu, H., 2025b. Opendrive-vla: Generalist vision-language-action agent for autonomous driving. arXiv preprint arXiv:2505.01871 URL: https://arxiv.org/abs/2505.01871.

Zhu, Y., Gupta, A., Ebert, F., et al., 2020. robosuite: A modular simulation framework and benchmark for robot learning. arXiv preprint arXiv:2009.12293 .

Zitkovich, B., Yu, T., Xu, S., Xu, P., Xiao, T., Xia, F., Wu, J., Wohlhart, P., Welker, S., Wahid, A., et al., 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. Conference on Robot Learning, PMLR , 2165–2183.

Table 6
Vision Encoder Scale Categories

Category	Log_{10}(Params)	Size (M)	Representative Models / Characteristics
Small	< 7.3	< 20	ResNet-18, MobileNet, ViT-Tiny; low-latency perception
Medium	$[7.3, 8.3)$	20–200	ResNet-50, ViT-Base, CLIP-RN50; strong semantic grounding
Large	≥ 8.3	> 200	ViT-Large, CLIP-ViT-L, ViT-G/14, SAM; high-level multimodal alignment

Table 7
Language Encoder Scale Categories

Category	Params (B)	Representative Models	Description
Small	< 1	T5-Base, MiniLM, DistilBERT, CLIP-Text	Basic grounding for short commands
Medium	1–10	LLaMA-2-7B, Qwen-7B, Gemma-7B, Mistral-7B	Semantic reasoning and contextual interpretation
Large	> 10	GPT-3.5/4, PaLM-2, Qwen-72B, LLaMA-70B	Advanced reasoning and generalization across tasks

Appendices

A. Key Terminologies

This appendix describes the computational procedures used to generate the key analytical figures in the main text. We begin by defining key terminology and then detail the statistical methods for each analysis. All analyses were conducted using Python 3.10 with NumPy, pandas, scikit-learn, statsmodels, and seaborn.

A.1. Key Terminology and Definitions

Before presenting the computational procedures, we define the key variables and concepts used throughout our analysis. The following terminology ensures consistency across regression, factor, and scale analyses.

Model Architecture Variables.

- **Fusion Depth (D_f):** Represents the stage in the network where visual and language features are combined: *Early Fusion* ($D_f = 1$): Features merged in the initial layers, enabling joint representation learning but potentially losing modality-specific detail. *Late Fusion* ($D_f = 2$): Features merged at the final decision layers, preserving modality-specific processing before integration. *Hierarchical Fusion* ($D_f = 3$): Multi-level integration across multiple network depths, allowing progressive refinement of cross-modal representations.

- **Decoder Family:** Defines the type of action prediction mechanism: *Autoregressive:* Sequential action prediction (e.g., Transformers, LSTMs). *Diffusion:* Iterative denoising-based action generation. *Flow:* Normalizing flow-based action modeling. *MLP:* Direct feedforward prediction from latent representations. *Planner:* Explicit optimization or search-based planning algorithms.

- **Encoder Families:** *Vision Encoder Family:* Processes visual inputs (e.g., CLIP-based, ViT/Transformer, ResNet, EfficientNet, DINO-based, SigLIP, CNN-based). *Language Encoder Family:* Processes textual or semantic inputs (e.g., BERT, GPT, T5, LLaMA, Qwen, PaLM, Gemini, CLIP-Text).

- **Model Scale Variables.** The vision and language encoders were categorized into three sizes explained in the table below.

- **Task Complexity Variables. Task Complexity (C_{task}):** Composite metric capturing task difficulty, diversity, and reasoning requirements. Higher values correspond to more challenging or multi-stage manipulation tasks. **Modality Richness (C_{mod}):** Number of distinct input modalities: 2 for vision + language, 3 for vision + language + proprioception, and 4 for vision + language + proprioception + tactile. **Dataset Size (N):** Number of training samples, transformed as: $\text{LogN} = \log_{10}(N)$

- **Performance Metrics. Normalized Success (Y):** Normalized task success rate (0-1 scale), adjusted for task difficulty. This is the primary dependent variable used in regression analyses. **Generalization Index:** Quantifies model performance on novel or unseen scenarios (0-1 scale), reflecting generalization ability beyond the training distribution.

B. Statistical Methodologies and Key Concepts

- **Standardization (z-score normalization):** Transformation of variables to zero mean and unit variance:

$$\tilde{x} = \frac{x - \mu}{\sigma} \tag{16}$$

where μ is the mean and σ is the standard deviation. This ensures coefficients are comparable across different scales.

- **Coefficient (β):** In regression, represents the expected change in the outcome variable for a one-unit increase in the predictor, holding other variables constant.

- **Ordinary Least Squares (OLS) Regression:** A statistical method for estimating the relationship between a dependent variable (outcome) and one or more independent variables (predictors) by minimizing the sum of squared residuals. OLS finds the best-fitting linear model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon \tag{17}$$

where Y is the outcome, X_i are predictors, β_i are coefficients to be estimated, and ε is the error term. The method assumes: Linear relationship between predictors and outcome, Independence of observations, Homoscedasticity (constant variance of errors), and Normally distributed errors

- **Forest Plot:** A graphical display of regression coefficients and their confidence intervals, arranged vertically to facilitate comparison. Each coefficient is represented by: A point estimate showing the β value, A horizontal line showing the 95% confidence interval, and A vertical reference line at zero.
Coefficients whose confidence intervals do not cross zero are statistically significant at $p < 0.05$. Forest plots are particularly useful for: Comparing effect sizes across multiple predictors, Identifying significant vs. non-significant effects at a glance, Visualizing uncertainty in parameter estimates, and Grouping related predictors thematically

- **Factor Analysis:** A dimensionality reduction technique that identifies underlying latent factors explaining correlations among observed variables. The method assumes observed variables are linear combinations of unobserved factors:

$$X_i = \lambda_{i1} F_1 + \lambda_{i2} F_2 + \cdots + \lambda_{ik} F_k + u_i \tag{18}$$

where: X_i is the i-th observed variable (standardized), F_j are latent factors (unobserved constructs), λ_{ij} are factor loadings (correlations between X_i and F_j), u_i is unique variance (not explained by factors), k is the number of factors (typically $k \ll p$, where p is number of variables).

Factor analysis helps answer questions like: *What underlying constructs explain why certain VLA design features tend to co-occur?* For example, vision and language model sizes may load on a common *Model Scale* factor, while fusion depth and fusion type may load on an *Architecture Design* factor.

- **Factor Loadings:** Correlations between observed variables and latent factors, ranging from −1 to +1. High loadings ($|\lambda_{ij}| > 0.4$) indicate strong associations: $\lambda_{ij} > 0.4$: Variable X_i increases with factor F_j, $\lambda_{ij} < -0.4$: Variable X_i decreases with factor F_j, and $|\lambda_{ij}| < 0.4$: Weak or negligible association.

B.1. Data Preprocessing

Our dataset includes: fusion depth strategy, fusion type, decoder family, application domain, vision encoder parameters (V_{params}), language model parameters (L_{params}), task complexity (C_{task}), modality richness (C_{mod}), dataset size (N), and adjusted success rate (Y).

Continuous features were log-transformed to address skewness:

$$\text{VisionParams} = \log_{10}(V_{\text{params}}), \quad \text{LLMParams} = \log_{10}(L_{\text{params}}), \\ \text{LogN} = \log_{10}(N) \tag{19}$$

Fusion depth was ordinally encoded (1=early, 2=late, 3=hierarchical), and categorical variables were one-hot encoded. All continuous predictors were z-score normalized ($\tilde{x} = (x - \mu)/\sigma$) prior to regression analysis.

B.2. OLS Regression and Forest Plot

We estimated the following ordinary least squares model:

$$\begin{aligned} Y = {} & \beta_0 + \beta_1 D_f + \beta_2 S_v + \beta_3 S_\ell + \beta_4 C_{\text{task}} + \beta_5 C_{\text{mod}} \\ & + \beta_6 \mathbb{1}_{\text{diffusion}} + \beta_7 \mathbb{1}_{\text{flow}} + \beta_8 \mathbb{1}_{\text{hierarchical}} \\ & + u_{\text{bench}} + \varepsilon. \end{aligned} \tag{20}$$

where Y is the normalized success rate, D_f is ordinal fusion depth, and $\mathbb{1}$ terms are binary indicators for decoder type and fusion strategy. The coefficients were estimated using maximum likelihood and the 95% confidence intervals were calculated using the t-distribution with $n - p$ degrees of freedom.

The forest plot (Fig. 10) displays standardized coefficients $\hat{\beta}_j$ with their confidence intervals $[\hat{\beta}_j \pm t_{0.975} \cdot \text{SE}(\hat{\beta}_j)]$. The coefficients are grouped thematically: *Architecture Design* (D_f, $\mathbb{1}_{\text{hier}}$), *Model Scale* (VisionParams, LLMParams), *Task Complexity* (C_{task}, C_{mod}), and *Decoder Policy* ($\mathbb{1}_{\text{diff}}$, $\mathbb{1}_{\text{flow}}$). A vertical reference line at $\beta = 0$ facilitates the assessment of statistical significance.

B.3. Factor Analysis

We perform exploratory factor analysis on six continuous features (D_f, VisionParams, LLMParams, C_{task}, C_{mod}, LogN) to identify latent constructs. The factor model assumes:

$$\mathbf{X} = \mathbf{L}\mathbf{F}^T + \mathbf{\Psi} \tag{21}$$

where $\mathbf{X}$ is the standardized characteristic matrix, $\mathbf{F}$ contains latent factor scores, $\mathbf{L}$ is the loading matrix and $\mathbf{\Psi}$ represents unique variances.

We extracted $k = 3$ factors using maximum likelihood estimation via expectation-maximization, explaining approximately 75% of total variance. The three factors were interpreted as:

- **Factor 1 (Architecture)**: High loadings on fusion depth
- **Factor 2 (Scale)**: Dominated by VisionParams and LLMParams
- **Factor 3 (Performance)**: Primarily loaded on dataset size and task complexity

The heatmap (Fig. 12) visualizes the 3×6 loading matrix $\mathbf{L}^T$ using a diverging colormap centered at zero, with annotated values. Loading magnitudes $|\lambda_{ij}| > 0.4$ indicate substantial associations between features and factors.

B.4. Software and Reproducibility

Analysis pipeline: Python 3.10.12, NumPy 1.24.3, pandas 2.0.3, scikit-learn 1.3.0, statsmodels 0.14.0, matplotlib 3.7.2, seaborn 0.12.2. Complete code and data are available in supplementary materials with fixed random seeds (seed=0) for reproducibility.

B.5. Limitations

Key limitations include: (1) linearity assumptions in OLS may not capture complex interactions; (2) models are not strictly independent due to incremental improvements; (3) measurement error in reported parameter counts; (4) publication bias toward successful models; (5) temporal confounding from hardware/data advances. Despite these limitations, robust comparative insights into VLA design principles were obtained.

C. Theoretical Entropy and Fusion Energy Computation

This appendix provides detailed mathematical derivations and computational procedures for the theoretical quantities visualized in the analysis of VLA fusion theory (Fig. 13). These metrics quantify the information-theoretic properties of multimodal fusion in Vision-Language-Action models.

C.1. Theoretical Framework

Vision-Language-Action models integrate information from multiple modalities (vision, language, proprioception) to generate robot actions. We model this fusion process using information theory principles, focusing on three theoretical quantities:

1. ΔH_k – Entropy reduction (uncertainty reduction from fusion)
2. η_k – Cross-modal attention efficiency (computational cost-benefit ratio)
3. $E_{\textbf{fusion}}$ – Fusion energy (total information processing capacity)

C.2. Entropy Reduction (ΔH_k)

Definition: Entropy reduction measures how much task uncertainty is eliminated by the VLA model. Higher entropy reduction indicates more effective uncertainty resolution. For a robotic task with initial uncertainty H_0 and post-fusion uncertainty H_k:

$$\Delta H_k = H_0 - H_k \tag{22}$$

In our empirical analysis, we use the **complementary success rate** as a proxy for residual uncertainty:

$$\Delta H_k = 1 - \text{Success}_{\text{rate}} \tag{23}$$

Initial uncertainty H_0 is normalized to 1 (maximum uncertainty before fusion). Residual uncertainty $H_k \approx (1 - \text{Success}_{\text{rate}})$ represents unresolved task ambiguity. Lower success rates → higher residual uncertainty → lower entropy reduction. Higher success rates → lower residual uncertainty → higher entropy reduction.

C.3. Cross-Modal Attention Efficiency (η_k)

Definition: Attention efficiency quantifies the information gain per unit of computational cost. It measures how efficiently the model leverages multimodal complexity relative to parameter budget. Mathematically,

$$\eta_k = \frac{C_{\text{mod}} \times C_{\text{task}}}{S_v + S_\ell} \tag{24}$$

where, C_{mod} is Modality richness (number of input modalities), C_{task} represents Task complexity score (difficulty, diversity, generalization), S_v is Vision encoder size ($\log_{10}$ of parameter count), and S_ℓ is Language model size ($\log_{10}$ of parameter count).

Interpretation: The representational demand, expressed as the numerator ($C_{\text{mod}} \times C_{\text{task}}$), captures the interaction between modality richness and task difficulty; higher values indicate tasks requiring more complex multimodal reasoning. The denominator (S_v+S_ℓ) represents the computational capacity, calculated as the log-scaled sum of vision and language model parameters, reflecting the total parameter budget available for multimodal processing. The resulting efficiency ratio η_k quantifies how effectively a model converts capacity into performance: high η_k values denote efficient models that achieve sophisticated reasoning with fewer parameters, while low η_k values indicate architectures that rely on large models to solve relatively simple tasks.

C.4. Fusion Energy (E_{fusion})

Definition: Fusion energy represents the total information-processing capacity mobilized during multimodal integration. It combines success rate, model scale, and modality richness into a unified metric. Mathematically;

$$E_{\text{fusion}} = \text{Success} \times (S_v + S_\ell) \times \ln(1 + C_{\text{mod}}) \quad (25)$$

where, Success is adjusted success rate (0-1), $S_v + S_\ell$ is Total model capacity (log-scale parameters), $\ln(1 + C_{\text{mod}})$, Modality complexity (logarithmic scaling).

D. Domain and Component Analysis Methodology

This appendix provides detailed computational procedures for the domain-wise and component-wise performance analysis figures. These analyses examine how VLA performance varies across application domains (manipulation, navigation, humanoid, GUI) and architectural components (decoders, vision encoders, language models). All computations follow the same data preprocessing pipeline described in Appendix A.

All analyses follow consistent statistical aggregation (group means, standard deviations) and filtering criteria ($n \geq 3$, exclusion of ambiguous categories). Visualization employs perceptually uniform color palettes, outside legends, error bars, and domain-specific background shading for the 8-panel analysis.

These analyses complement the regression-based forest plot (Appendix A) and entropy-based theoretical analysis (Appendix C) by providing descriptive, exploratory perspectives on VLA performance heterogeneity.

D.1. Decoder Analysis

Purpose: Evaluate how different action decoder architectures impact task success and generalization capability. *Data Grouping:* Models are grouped by `DecoderFamily`: **Autoregressive**: Sequential action prediction (e.g., transformers, LSTMs), **Diffusion**: Iterative denoising-based action generation, **MLP**: Direct feedforward prediction, **Planner**: Explicit planning algorithms, **Flow**: Normalization of flow-based modeling (excluded if $n < 3$)

Filtering Criteria: We exclude decoder families with fewer than 3 models to ensure statistical reliability:

Statistical Aggregation: For each decoder family, we compute the following:

$$\bar{Y}_d = \frac{1}{n_d} \sum_{i \in \mathcal{D}_d} Y_i \quad (26)$$

where, $\bar{Y}_d$ is the mean success rate for decoder family d, n_d is the number of models with decoder d, $\mathcal{D}_d$ is the set of models that uses the decoder d, and Y_i is the success rate for model i

Standard deviation across models in each family:

$$\sigma_d = \sqrt{\frac{1}{n_d - 1} \sum_{i \in \mathcal{D}_d} (Y_i - \bar{Y}_d)^2} \quad (27)$$

Standard error of the mean:

$$\text{SE}_d = \frac{\sigma_d}{\sqrt{n_d}} \quad (28)$$

D.2. Encoder Analysis

Purpose: Evaluate how vision and language encoder choices affect performance and generalization. Data Grouping:

Vision Encoder Families: CLIP-based (e.g., CLIP-ResNet50, CLIP-ViT), ViT/Transformer (e.g., ViT-Base, ViT-Large), ResNet (e.g., ResNet-18, ResNet-50), EfficientNet (e.g., EfficientNet-B3), DINO-based (e.g., DINOv2), and SigLIP.

Language Model Families: BERT (e.g., BERT-base), GPT (e.g., GPT-3.5, GPT-4), T5 (e.g., T5-Small, T5-Base), LLaMA (e.g., LLaMA-2-7B, LLaMA-70B), Qwen (e.g., Qwen-7B, Qwen-72B), PaLM (e.g., PaLM-2), Gemini/Gemma, CLIP Text, and Vicuna

D.3. Domain Analysis

Purpose: Compare VLA performance across different application domains to identify domain-specific challenges.

Domain Categories.

- **Manipulation**: Pick-and-place, assembly, tool use (e.g., kitchen tasks, object rearrangement)
- **Navigation**: Mobile robot navigation, path planning (e.g., indoor navigation, obstacle avoidance)
- **Humanoid**: Whole-body control for humanoid robots (e.g., bipedal walking, reaching)

Statistical Aggregation: For each domain d, compute mean and standard deviation as in previous sections.

D.4. Domain-Component Analysis

Purpose: Joint analysis of how architectural components (decoders, encoders) perform across different application domains. This reveals domain-specific design principles.

Statistical Computation: For each panel showing domain × component interaction:

$$\bar{Y}_{d,c} = \frac{1}{n_{d,c}} \sum_{i \in \mathcal{D}_{d,c}} Y_i \quad (29)$$

where: $\bar{Y}_{d,c}$ is mean success for domain d and component c, $n_{d,c}$ is number of models with domain d and component c, and $\mathcal{D}_{d,c}$ is the set of such models

These values are organized into a pivot table (domain matrix × component) to plot:

E. Data and Code Availability

Data that we used to perform the above analysis and the complete code for the plots are available as supplementary material. Upon acceptance, we will provide the GitHub repository with all the supplementary material.