Learning Physical Interaction: A Survey of Tactile- and Force-aware Robot Learning

Shilin Shan[1,*,‡], **Chuhao Zhou**[1,*], **Ruize Wang**[1,*], **Xinyan Chen**[1,*], **Xiangyu Chen**[1,*], **Xinyu Zhou**[1,*], **Boyu Ma**[1,*], **Iris Yuxuan Hu**[1], **Jingliang Li**[1], **Celeste Yuxuan Hu**[1], **Geng Li**[1], **Guohao Chen**[1], **Tianrui Zhu**[1], **Zhe Li**[1], **Yanjie Ze**[2], **Haoran Geng**[3], **Zhiyang Dou**[4], **Jianxin Bi**[5], **Yuejiang Liu**[2], **Jianshu Zhou**[5], **Jiachen Li**[6], **Paul Liang**[4], **Tatsuya Harada**[7], **Robert Katzschmann**[8], **Harold Soh**[5], **Na Li**[9], **Edward Johns**[10], **Danica Kragic**[11], **Jan Peters**[12], **Wojciech Matusik**[4], **Masayoshi Tomizuka**[3], **Jitendra Malik**[3], **Jianfei Yang**[1,†]

[1]Nanyang Technological University, [2]Stanford University, [3]University of California, Berkeley,
[4]Massachusetts Institute of Technology, [5]National University of Singapore, [6]Georgia Institute of Technology,
[7]The University of Tokyo, [8]ETH Zurich, [9]Harvard University, [10]Imperial College London,
[11]KTH Royal Institute of Technology, [12]Technical University of Darmstadt
[*]Equal Contribution, [‡]Project Lead, [†]Corresponding Author

Physically grounded robot intelligence requires robots to perceive, reason about, and regulate their interactions with the physical world. This capability is particularly critical in contact-sensitive manipulation, where successful task execution depends not only on visual perception and motion generation, but also on force regulation and adaptive control. In this context, recent robot learning methods have made substantial progress by integrating force, tactile, vision, language, and proprioceptive sensing into learned manipulation policies. In parallel, many systems adopt multi-phase architectures that combine high-level policies, action-refinement modules, and low-level controllers to bridge semantic task understanding with reactive physical execution. Despite these advances, existing surveys have not explicitly reviewed force- and tactile-aware robot learning from a unified perspective that jointly captures multimodal sensing and multi-phase system design. This survey addresses this gap by proposing **TF-ART**, a **T**actile/**F**orce-**A**ware **R**obot learning **T**axonomy for multimodal and multi-phase frameworks, which maps individual methods into a unified hierarchical structure. The framework characterizes how recent works organize observation modalities, encode and fuse heterogeneous sensory inputs, generate and refine actions across multiple phases, and connect learned policies to reactive robot-end control. Building on this methodological view, we further examine the task settings and infrastructure requirements of physical interaction, thereby integrating both algorithmic and practical perspectives on force- and tactile-aware robot learning.

Keywords: Force-aware Manipulation, Multimodal Learning, Compliant Control, Embodied AI
GitHub: https://github.com/NTUMARS/Awesome-Tactile-Force-aware-Robot-Learning
Webpage: https://lorenzo-0-0.github.io/tactile-force-survey/
Correspondence: Jianfei Yang at jianfei.yang@ntu.edu.sg;

1 Introduction

1.1 Background

Embodied intelligence has experienced rapid development in recent years, driven by advances in multimodal sensing, large-scale reasoning models, and generative motion planning [54, 55, 56, 57, 58]. Existing approaches often combine visual perception with language instruction for object recognition and scene understanding. These observations provide rich semantic and geometric cues and are effective for general manipulation tasks. However, when robots move from free-space motion to physical interaction, the most task-critical information often shifts from visual observations to interaction feedback [11, 18, 30, 36]. Under contact, the target object may be occluded, its state may change due to deformation, and successful execution may depend on where contact occurs, whether it is stable, and how large the interaction force becomes. Force and tactile sensing therefore provide essential feedback that complements vision and proprioception [5, 10, 21, 59]. Specifically,

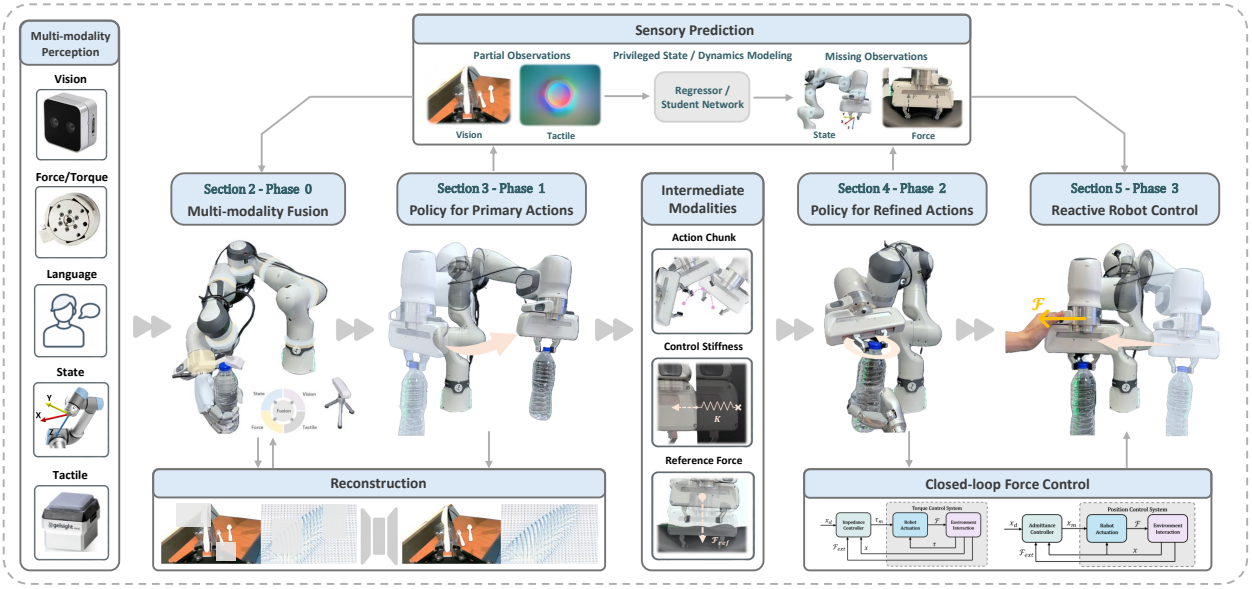

Figure 1 Overview of TF-ART: a unified force/tactile-aware multi-modal multi-phase framework. The main hierarchical architecture includes multimodal perception, encoding and fusion, primary action generation, action refinement, and low-level robot-end control. The branching modules include input data reconstruction, missing-modality prediction, and explicit force-control algorithms.

force sensing captures global interaction information, such as end-effector wrench and joint torque, supporting force regulation and compliant control. Tactile sensing captures local contact information, such as pressure distribution, deformation, and surface texture. Together, force and tactile feedback allow robots to sense, interpret, and regulate physical interaction beyond what can be inferred from vision-centered observations.

The use of force and tactile information has been studied from both learning-based and control-oriented perspectives. Learning-based methods enable task-relevant representations and action-generation policies to be developed from collected data, including expert demonstrations and autonomous interaction data gathered through exploration, rather than requiring their behavior to be fully specified analytically [5, 31, 32, 36]. Classical control-oriented methods use explicitly designed feedback mechanisms to provide stable physical interaction and force regulation under contact [60, 61, 62, 63]. These approaches offer different but compatible ways of developing contact-aware robot behavior, and recent literature increasingly combines learning-based and control-oriented components across different parts of the manipulation pipeline. This development is reflected in two closely related directions: multimodal integration for contact-aware perception and action generation, and multi-phase policy-to-control architectures for contact-sensitive execution.

From the perspective of multimodal integration, recent works have explored the complementarity between contact-free and contact-rich perception by jointly integrating force, tactile, vision, language, and proprioceptive state information for action generation [23, 24, 33, 46, 59]. Since the same object or interaction event can be observed through multiple sensory modalities, multimodal representation learning provides a natural way for different modalities to exchange information. These works therefore raise key questions about how force and tactile signals complement other modalities, how their representations can leverage existing pre-trained models, and how such integration guides contact-aware action generation.

From the perspective of multi-phase design, force and tactile feedback are increasingly incorporated across different stages of the action-generation pipeline rather than treated as passive inputs to a single end-to-end policy. In recent studies, a high-level policy may first generate a coarse trajectory from vision, language, and state observations, while subsequent phases use force or tactile feedback for motion refinement or direct force control [11, 16, 45]. This pattern highlights a central question in contact-rich manipulation: semantic reasoning and global trajectory generation can benefit from large vision-language-based policies, yet such policies may not satisfy the low-latency and high-precision requirements of local, force-sensitive interaction.

Table 1 Comparison of prior surveys by their primary organizing dimensions.

Survey family	Representative surveys	Obs.	Rep./Fusion	Policy	Refine.	Control	Pipeline
Tactile sensing, skins, and hardware	[64, 65, 66, 67, 68, 69, 70, 71, 72, 73]	•	○	–	–	○	–
Tactile and visuo-haptic perception	[74, 75, 76, 77, 78]	•	•	○	–	–	–
Active-haptic and sensorimotor interaction	[79, 80]	•	○	○	○	•	○
General robot learning and learning from demonstration	[81, 82, 83, 84, 85]	○	○	•	–	○	–
Contact manipulation and contact-rich imitation learning	[86, 87]	○	○	•	○	•	○
Dexterous and interactive robot learning	[88, 89, 90]	○	○	•	○	○	–
Robot foundation-model perspectives	[91, 92, 93]	○	•	•	○	○	○
Broad tactile-robotics outlook	[94]	•	○	○	○	○	–
TF-ART (ours)	**This survey**	•	•	•	•	•	•

Obs.: contact-aware multimodal observations; *Rep./Fusion*: multimodal representation and fusion; *Policy*: learned primary action generation; *Refine.*: contact-aware action refinement; *Control*: robot-end force/compliance control; *Pipeline*: unified policy–control pipeline mapping. • denotes a primary organizing dimension, ○ denotes partial coverage, and – denotes no systematic coverage.

Taken together, the above developments suggest that force- and tactile-aware robot learning is no longer defined merely by the inclusion of additional sensory inputs, but by how these inputs are integrated with other modalities and organized across multiple stages of the policy pipeline. Despite this progress, existing research tends to approach multimodal learning and multi-phase design from only partial perspectives, usually covering only a subset of components such as representation learning, multimodal fusion, policy architecture, action refinement, and robot-end control. Although prior surveys have covered related topics [81, 82, 86, 87], as summarized in the next subsection, these discussions remain scattered across different areas and do not yet provide a unified view of contact-rich manipulation. This gap motivates a dedicated survey on force- and tactile-aware robot learning.

1.2 Related Surveys

Earlier tactile surveys primarily organized the field around sensing technologies and embodiment. Foundational reviews examined tactile transduction, sensor structures, signal processing, and robotic applications [64, 65, 66, 67]. Subsequent surveys extended this perspective to dexterous hands, tactile skins, application-specific sensing requirements, and soft robotic skins [68, 69, 70, 71, 72], while Cheng et al. [73] considered the integration of robot skin with sensing, control, and applications. Other reviews shifted the focus toward tactile perception and representation [74, 75, 78], visuo-haptic fusion [76, 77], and active sensorimotor interaction [79, 80]. Together, these surveys establish the sensing and perception foundations of tactile robotics, but are generally organized around sensors, representations, or specific interaction capabilities rather than learned policy pipelines.

A complementary body of surveys focuses on robot learning and contact-rich manipulation. General reviews cover manipulation learning and learning from demonstration [81, 82, 83, 84], generative robot learning [85], and foundation models [91, 92]. Surveys of contact manipulation and contact-rich imitation learning discuss task formulations, demonstrations, learning methods, and force/compliance control [86, 87], while dexterous and interactive-learning surveys emphasize hand hardware, teleoperation, and human involvement [88, 89, 90]. Particularly close to our scope, the tactile-robotics outlook [94], the sensorimotor reviews [79, 80], and the force-aware foundation-model perspective [93] cover several components of force- and tactile-aware learning. However, none uses the complete multimodal policy-control pipeline as its primary organizing axis.

Table 1 compares the evaluation dimensions of related surveys. TF-ART differs from prior surveys mainly in its unit of analysis. Rather than organizing methods solely by sensor type, task, or learning paradigm, TF-ART maps each method onto a common information and control flow: multimodal observation, representation and fusion, primary action generation, contact-aware refinement, and robot-end control. This organization identifies not only whether force or tactile feedback is used, but also where and for what purpose it enters the system. TF-ART thereby connects multimodal integration with multi-phase policy organization and provides a unified view of how learned policies interface with high-frequency force and compliance control.

1.3 Contributions

Motivated by the need for a unified organization of force- and tactile-aware robot learning, this survey proposes **TF-ART**, a **T**actile/**F**orce-**A**ware **R**obot learning **T**axonomy for multimodal and multi-phase frameworks, as illustrated in Figure 2. The main contributions are summarized as follows:

- TF-ART summarizes recent methodologies through a unified hierarchical architecture. By mapping diverse methodological pipelines to corresponding framework modules, TF-ART provides a comprehensive view of recent progress in force- and tactile-aware robot learning.
- We examine how force and tactile signals are incorporated into multimodal robot policies. Beyond listing input modalities, we discuss how contact-related observations are represented, aligned, and fused with vision, language, and state information, as well as how different primary policy architectures use these fused representations for action generation. The discussion emphasizes the basic mechanisms, benefits, limitations, and task-dependent suitability of these multimodal designs.
- We analyze how force and tactile feedback reshape robot learning pipelines beyond single-stage action prediction. In particular, we discuss phase-specific sensing, discriminative modality injection, action refinement, and low-level control, showing how recent systems use contact feedback to correct high-level policy outputs, improve real-time responsiveness, and support stable physical interaction.
- We discuss practical and implementation-level factors in force- and tactile-aware robot learning. These include sensor types and sensor models, robot hardware capabilities, and representative contact-rich manipulation tasks considered in recent studies.

1.4 Framework Overview and Survey Organization

Figure 2 presents the complete TF-ART framework and illustrates how the reviewed force- and tactile-aware studies are mapped onto it. Each paper is analyzed and categorized according to its methodological components, allowing its pipeline to be aligned with the corresponding modules of the proposed framework. Specifically, in a top-down sequence, TF-ART organizes its components as follows: (a) the observation space, that is, the sensory measurements used as model inputs; (b) input encoding and modality fusion methods, grouped according to modality-usage criteria; (c) input reconstruction and representation methods for encoder training; (d) primary action-generation architectures, which constitute the phase-one policy and are shared by most reviewed works; (e) sensory prediction methods for missing-modality replacement; (f) intermediate modalities that cannot be directly measured and are better suited for control or action refinement; (g) the action-refinement layer, or phase-two policy, which leverages forwarded encoded inputs and phase-one predictions to generate refined actions for fine-grained objectives; and (h) the control layer, or phase three, which connects learned models to robot hardware and supports safe execution through explicit force-control mechanisms.

For literature search and selection, we consider a method tactile-aware when it uses contact measurements from camera-based tactile sensors, tactile pads or arrays, robotic skins, or related contact-sensing devices. A method is considered force-aware when it uses or explicitly estimates force-related information obtained from wrist or end-effector force/torque sensors, joint-torque sensors, motor currents, or other proprioceptive signals. To qualify, tactile or force information must be used as a policy input or training signal, or explicitly predicted as an intermediate representation or output within the learned action-generation process. Robot learning is defined here as a data-driven method that generates robot actions or action-related control targets. Accordingly, we exclude studies limited to sensor design, calibration, contact or force estimation, and representation learning without evaluation in a downstream action-generation policy, as well as classical controllers without a learned action-generation component and generic manipulation policies that neither use nor predict tactile or force information. Multi-phase design is an analysis category rather than an inclusion requirement; it refers to an explicit separation between primary action generation and one or more subsequent contact-aware refinement or force/compliance-control stages. Candidate papers were identified through searches of Google Scholar, arXiv, and IEEE Xplore, supplemented by backward and forward citation tracing. The search window for the core corpus mapped to TF-ART covered work published from 2024 through April 2026, while the survey's broader reference list was last updated on July 21, 2026. This process yielded a final TF-ART corpus of 53 papers.

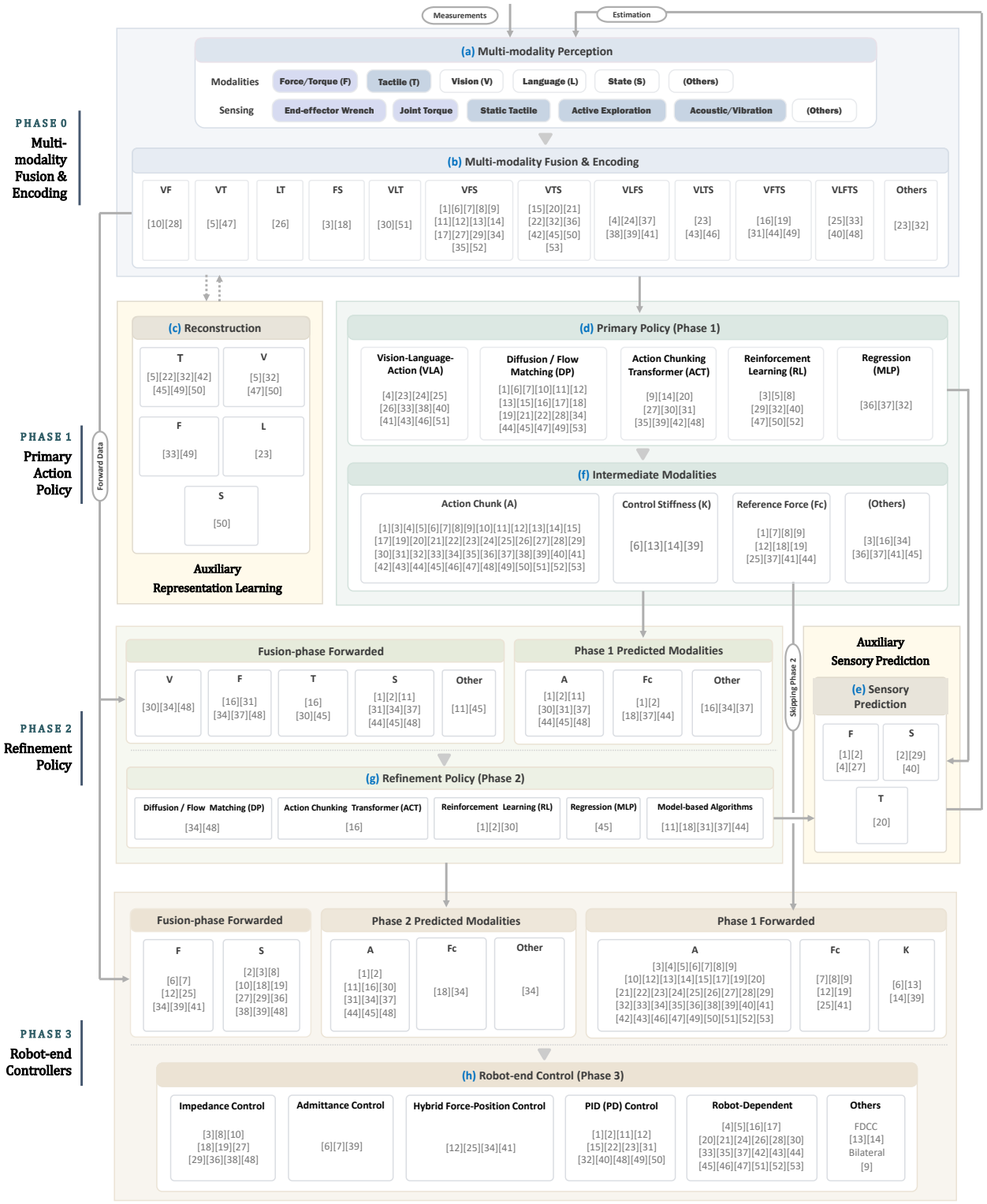

Figure 2 Detailed TF-ART framework covering all investigated research papers.* The framework components span the following: (a) multimodal observation; (b) multimodal fusion; (c) input reconstruction; (d) Phase 1: primary action-generation policy; (e) sensory prediction; (f) intermediate modalities; (g) Phase 2: action-refinement policy; and (h) Phase 3: robot-end control.

*Readers may use the PDF viewer's search function to locate a paper's reference number, as listed in the reference section, within the framework and inspect the corresponding components and methodological architecture categorized in this survey.

This survey is organized around the proposed unified framework, with each section corresponding to a major component of the overall architecture. This organization is also reflected in Figure 2. Section 2 introduces the modalities commonly used in recent works, how they are encoded and fused during training, and how encoders can be trained through input data reconstruction. Section 3 discusses policy architectures for primary action generation, following mainstream research directions, together with intermediate modality prediction for the next phase and sensory prediction for missing measurements. Section 4 discusses techniques for refining actions generated by phase-one policies, as well as modality-selection considerations for fast inference. Section 5 introduces the fundamentals of model-based low-level robot control, with a particular focus on force-adaptive and compliance-control techniques used for both data collection and real-world interaction. Section 6 discusses common tasks and real-world settings demonstrated in existing studies, including their characteristics and practical implications.

2 Phase 0: Multimodal Fusion and Representation Learning

A multimodal robotic system must address three key questions: which sensing modalities are needed for a given task, how raw multimodal observations can be represented and grounded in robotic control, and how complementary information across modalities can be effectively fused and exploited. Accordingly, this section reviews prior works from five perspectives: tactile (Section 2.1), force (Section 2.2), non-contact modalities (Section 2.3), multimodal representation learning (Section 2.4), and multimodal fusion (Section 2.5). The surveyed papers are summarized in Table 2 according to the metrics introduced in this section. A visual illustration of the modalities, representation learning methods, and input-output data flow is shown in Figure 3. We refer to this part as Phase 0 as it defines the sensory and representational basis before action generation, whereas Phases 1-3 correspond to explicit stages of action generation.

2.1 Tactile (T)

2.1.1 Tactile Sensing Methods

Tactile perception during manipulation is closely coupled with action. The same object can produce different sensor readings when the contact location, normal force, motion direction, speed, or exploratory action changes. By controlling these variables, the robot can make different geometric and material properties easier to detect [95, 96]. Existing tactile sensing methods can be grouped into four categories according to how tactile signals are collected.

(a) Static contact and pressure sensing. When the robot maintains stable contact with little relative motion, the interaction produces a sustained deformation, pressure, or force distribution. The resulting observation may be represented as a tactile image, a pressure or force map, or a set of distributed contact measurements. Its spatial pattern reflects the contact location, area, and local surface geometry and can support the estimation of contact states and object attributes such as shape and pose. A single contact primarily captures a local surface patch, whereas multiple contacts distributed across the hand additionally capture the spatial relationships among different surface regions, facilitating object recognition and localization. These methods therefore rely primarily on static or quasi-static spatial contact patterns rather than exploratory motion or rapid temporal variations [74, 97].

(b) Active exploration of shape and material properties. Geometric and material properties often become clear only when the sensor moves in a suitable way. Rolling or sliding a fingertip over a surface converts cracks, bumps, and ridges into changes in the contact trajectory. Varying the normal force and speed produces different vibration and reaction-force patterns that help distinguish textures. When object identity or contact location is uncertain, the current estimate can also guide the next contact action [98, 99, 100]. Exploratory motion is therefore part of the sensing process rather than a fixed data-collection procedure.

(c) Dynamic contact events. Static pressure maps mainly describe sustained contact and may miss short-lived changes. Dynamic tactile sensing instead measures acceleration, vibration, or stress-rate signals to detect slip onset and fine surface features [101, 102]. Event-based tactile sensors encode rapid signal changes as asynchronous events and, when fused with event-based vision, support low-latency object classification and

rotational-slip detection [103]. In these methods, the temporal pattern of the signal is more important than the static pressure distribution.

(d) Tactile signal transmission through the hand, object, and tool. Contact signals can travel through the skin, hand, grasped object, or tool before reaching the sensor. The measured deformation and vibration therefore depend on the mechanical connection between the contact point and the sensor. Whole-hand vibration patterns and vibrations transmitted through tools can be used to infer contacts outside the immediate sensing area [104, 105, 106, 107]. This extends tactile perception beyond local sensing at the fingertip or skin.

Overall, tactile sensors can produce deformation images, taxel arrays, contact points, pressure or force sequences, vibration signals, and event streams. These measurements are used to estimate surface shape, texture, contact state, and interaction dynamics. Because the measurements change with the robot's actions, tactile perception naturally links sensing, exploration, and control.

2.1.2 Early Tactile Learning

Early tactile-learning methods used data-driven models to map sequences of tactile measurements and actions to object identity, material properties, contact states, or control commands. Representative methods can be grouped into three categories.

(a) Temporal recognition and continual learning. A single contact usually provides only local and ambiguous information. Temporal models reduce this ambiguity by combining measurements across a sequence of contacts. Online models update recognition when new measurements arrive, while incremental discriminative and generative models add new object classes without losing previously learned classes. Robust tactile descriptors further improve recognition across sensor types, exploratory motions, and interaction durations [108, 109, 110]. Temporal accumulation and online updating therefore make recognition less dependent on a single contact or a fixed training set.

(b) Multimodal learning from vision and touch. Vision and tactile sensing provide complementary information at different stages of manipulation. Vision captures global object geometry and scene context before contact, whereas tactile sensing reveals local contact conditions, deformation, and slip after contact. One use of this complementarity is to predict the outcomes of candidate grasp adjustments and select an appropriate regrasp action [111]. A broader formulation learns shared representations from paired visual and tactile observations through self-supervised learning before policy training, allowing the learned features to transfer across contact-rich tasks [112]. For interactions requiring faster responses, event-based methods fuse asynchronous visual and tactile streams for low-latency object classification and rotational-slip detection [103]. Together, these formulations use global visual context and local contact evidence for action-conditioned outcome prediction, transferable representation learning, or rapid contact-event detection.

(c) Tactile feedback for manipulation. Tactile feedback can be used at several levels of the control hierarchy. At the lowest level, contact and slip detection adjust grasp force. At the motion level, interaction data are used to learn compliant behaviors or maintain contact while following an object contour. At the skill level, manipulation is divided into primitives, and tactile states are used to monitor execution and trigger corrections or transitions [113, 114, 115, 116, 117]. This progression extends tactile learning from local grasp regulation to the execution of complete manipulation skills.

These studies established temporal modeling, uncertainty-based exploration, joint vision–touch learning, and closed-loop control as core parts of tactile learning. However, most systems were developed for a specific task, sensor, object set, robot platform, or exploration procedure. The methods reviewed in this survey build on these ideas through shared multimodal representations, broader task conditioning, and generative action models.

2.1.3 Tactile Measurement Format

To use different tactile sensors in a common robot-learning pipeline, raw sensor outputs must be converted into a representation that can be processed by the model. This survey considers three main forms that are commonly adopted in the surveyed 53 papers: visual-tactile images, tactile matrices, and contact points. They

differ mainly in how contact is organized spatially: as a dense image, a regular grid of sensing elements, or a sparse set of locations.

Visual-Tactile Images. Camera-based tactile sensors record changes in an elastomeric surface as images. Depending on the sensor design, an image may reflect contact shape, surface texture, marker displacement, and local deformation. After calibration, the image can also be converted into depth, displacement, force, shear, or deformation maps. Because these data follow a regular image grid, they can be processed directly using CNNs or Vision Transformers. This representation preserves fine spatial detail, but it is relatively high-dimensional and often depends on the sensor's optical design and calibration.

Tactile Matrices. A tactile matrix represents a regular array of tactile sensing elements, or taxels. Each grid cell corresponds to one taxel and may contain pressure, normal force, shear force, displacement, or vibration. CNNs or Transformers can process the grid while preserving the spatial relationships between neighboring taxels. Compared with camera-based tactile images, this representation can be lower-dimensional and often records physical quantities more directly. However, its shape and resolution are tied to the layout of the sensor.

Contact Points. Contact points are useful when active contacts are sparse or sensing elements are placed on an irregular surface. Each point stores a spatial location and may include attributes such as force, displacement, surface normal, or contact state. Point-based encoders or Transformers can process a variable number of points without requiring a fixed two-dimensional grid. This representation adapts well to curved surfaces and different sensor layouts, although sparse points may not retain fine texture or dense deformation patterns.

All three forms can also be arranged as temporal sequences to capture changes during contact. They are not mutually exclusive: tactile images can be converted into depth or force maps, taxel grids can be reduced to active contact points, and contact points can be combined with visual geometry. Together, they define the tactile input branch of Phase 0 and allow tactile measurements to be encoded and fused with vision, language, force/torque, and robot state.

2.2 Force/Torque (F)

2.2.1 Force/Torque Sensing Methods

Force/torque sensing provides information about external contact loads exerted on the robot by the environment or a human. These loads are typically represented as a six-dimensional wrench at the end effector or as torques transmitted through individual joints. They can be measured directly using dedicated sensors or estimated indirectly from robot states and actuator signals. This subsection reviews six-axis force/torque sensing, joint torque sensing, and sensorless external-load estimation.

(a) Six-axis force/torque sensing. A conventional six-axis force/torque sensor contains a slightly deformable elastic structure. When an external force or torque is applied, the structure bends or twists slightly. Strain gauges attached to the structure convert these deformations into changes in electrical quantities, such as resistance or voltage. Forces and torques applied along different directions produce distinct combinations of signals across the strain gauges. Using a mapping obtained through calibration, the sensor converts these signals into numerical measurements of wrench components [118, 119].

(b) Joint torque sensing. A joint torque sensor is installed inside a robot joint, typically between the gearbox and the connected link. It contains an elastic element that twists slightly as torque is transmitted through the joint. Strain gauges attached to this element convert the deformation into changes in electrical quantities, such as resistance or voltage. After calibration, these signals are mapped to a numerical torque measurement for the joint. The sensor is designed to respond primarily to torque about the joint axis while minimizing sensitivity to off-axis forces and moments [120, 121].

(c) Sensorless external-load estimation. When direct force/torque sensing is unavailable, contact information can instead be inferred from proprioceptive signals or visual observations. To fully replace dedicated force/torque sensors, proprioception-based approaches commonly train a separate estimator or explicitly model robot dynamics to estimate external contact loads. These approaches typically use indirect force-related signals, such as motor currents or joint-position tracking errors, model robot dynamics using either model-based

or learning-based methods, and estimate joint-level external torques or the contact wrench at the end effector [122, 123, 124, 125, 126, 127].

2.2.2 Force/Torque Measurement Format

This survey distinguishes three input forms according to where the load information is obtained: End-Effector Wrench, Joint Torque, and Tactile-Based Force Estimates.

End-Effector Wrench. An end-effector wrench combines the net force and moment acting at a specified reference frame. It is commonly represented as a 6-dimensional vector

$$\mathbf{w}_t = \begin{bmatrix} \mathbf{f}_t^\top & \mathbf{m}_t^\top \end{bmatrix}^\top \in \mathbb{R}^6,$$

where $\mathbf{f}_t \in \mathbb{R}^3$ is the Cartesian force and $\mathbf{m}_t \in \mathbb{R}^3$ is the Cartesian moment. The wrench may be measured directly using a wrist-mounted force/torque sensor or estimated from robot dynamics. Before being passed to a learning model, the signal is commonly bias-corrected, transformed into a consistent coordinate frame, and compensated for payload and gravity. A temporal encoder can then process either individual measurements or a short history of measurements to extract force/torque features.

Joint Torque. Joint torque is represented as a J-dimensional vector, where J is the number of actuated joints. Depending on the robot platform, these values may be measured by joint-torque sensors or estimated from motor currents and actuator models. Raw joint torque includes loads caused by both robot motion and external contact. Learning systems may therefore use the torque sequence together with joint position and velocity, or use an external-torque residual obtained after subtracting the robot's predicted internal loads. Joint-torque inputs are useful when interaction along the full robot arm is important or when a wrist-mounted force/torque sensor is unavailable.

Tactile-Based Force Estimates. Force can also be estimated from visual-tactile observations rather than measured using a dedicated force/torque sensor. A calibrated or learned estimator maps visual-tactile images to quantities such as local contact-force components or a 6D wrench [128, 129]. The resulting estimates can be represented as temporal sequences and processed in the same way as directly measured force signals. Unlike direct force/torque measurements, however, their accuracy depends on the tactile sensor design, calibration procedure, deformation model, and data used to train the estimator.

These three forms capture different aspects of physical interaction. An end-effector wrench summarizes the net load at a reference frame, joint torque describes loads at the robot's actuators, and tactile-based force estimation recovers force from local contact deformation. Together, they define the force/torque input branch of Phase 0 and allow load-related information to be encoded and fused with vision, language, tactile observations, and robot state for downstream action generation and control.

2.3 Non-contact Modalities

Beyond tactile and force/torque sensing, we further consider three complementary modality groups: language (L), vision (V), and proprioceptive/environmental state (S). We also briefly discuss other less commonly adopted modalities that may offer additional sensing and reasoning capabilities for future robotic systems.

2.3.1 Language (L)

Human Instructions are natural-language phrases or sentences that specify the task to be executed, e.g., "pick up an apple and place it on the table." These instructions are first tokenized into discrete textual units and converted into continuous text embeddings. A text encoder, such as BERT [130], then transforms these embeddings into language features that provide task-conditioning information for policy learning [131, 132, 133].

2.3.2 Visual Perception (V)

RGB Images are among the most important perception modalities in robot learning, as they provide rich appearance cues, including color, texture, object shape, and scene context. These images are typically represented by their height, width, and three color channels. A visual encoder, such as a Convolutional Neural Network (CNN) [134] or Vision Transformer (ViT) [135], then processes the raw image observations and converts them into visual features for policy-learning.

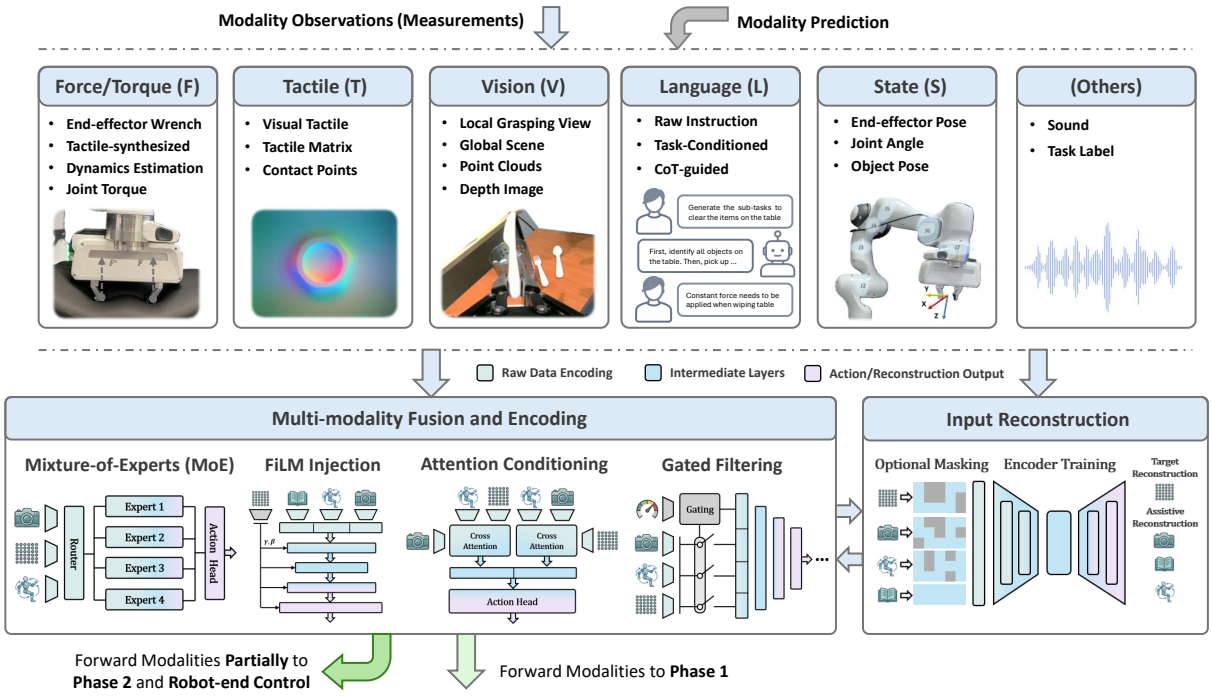

Figure 3 Overview of the multimodal fusion and encoding framework. The upper part visualizes the physical interpretation of different modalities and their sub-classes, while the lower part illustrates the model structures of major representation-learning and modality-fusion methods.

Depth Images provide explicit spatial information, with each pixel indicating the distance from the camera to the corresponding visible surface. Compared with RGB images, depth observations are less sensitive to appearance variations, making them useful for geometry-aware perception, grasping, collision avoidance, and spatial reasoning. Similar to RGB images, depth observations are typically processed by visual encoders such as CNNs or ViTs and converted into geometric visual features.

Point Clouds directly represent the 3D structure of objects and environments as a collection of spatial points, where each point records its position in 3D space. They are useful for spatial reasoning, object-pose estimation, affordance prediction, and manipulation in cluttered scenes. Point clouds are commonly processed by point-cloud encoders, such as PointNet [136] or Point Transformer [137], which convert the raw geometric observations into 3D structural features for policy-learning.

2.3.3 Proprioception and Environmental State (S)

EEF State measurement contains the 3D position of the end-effector, its orientation, and the gripper state in a given coordinate frame. The orientation is commonly described using Euler angles or quaternions, while the gripper state can be represented either as a binary open/closed indicator or as a continuous gripper width. EEF state is typically obtained from the robot's internal state feedback.

Joint Angle describes the robot configuration in joint space. A single measurement frame is represented as a vector of J joint-angle positions. Joint angles provide essential proprioceptive information for trajectory optimization and safe manipulation under kinematic constraints. They are usually read from the robot's internal state feedback.

Object Pose describes the 6D pose of an object in a camera or world frame. Following the common definition [138], each object pose consists of a 3D position and an orientation that specify the transformation from the object coordinate frame to a reference frame, such as the camera or world frame. Object pose provides compact task-relevant spatial information, enabling object-centric manipulation. These poses are commonly obtained from motion capture systems or vision-based 6D pose estimation methods such as FoundationPose [138].

Table 2 Summary of modality usage, fusion strategies, and representation learning in force- and tactile-aware robot learning methods.

Method	Input Modalities and Use Phase														Representation Learning	Fusion Method				
	Language (L)	Vision (V)			Force/Torque (F)			Tactile (T)			State (S)			Other						
	Instruction	Image	Depth	Point Cloud	EEF Wrench	Joint Torque	Tactile-synth.	Visual-Tactile	Tactile Matrix	Contact Points	EEF Pose	Joint Angle	Object Pose			FiLM	Attention Cond.	MoE	Gated Filter	Others
UPPFC [1]		P1			P1							P1, P2				✓	✓			
FCLM [2]												P2, P3					✓			
FORGE [3]					P1						P1, P3						✓			
TA-VLA [4]	P1	P1				P1						P1					✓			
M3L [5]		P1						P1							MAE		✓			
ACP [6]		P1			P1, P3						P1					✓	✓			
AdmitDiff [7]		P1			P1, P3						P1	P1				✓	✓			
HIL-SERL [8]		P1			P1						P1, P3						✓			
Bi-ACT [9]		P1				P1						P1					✓			
DexForce [10]		P1			P1						P3					✓	✓			
FoAR [11]		P1	P1	P1	P1, P2						P2					✓	✓		✓	
ForceMimic [12]		P1	P1	P1	P3								P1			✓	✓			
DIPCOM [13]		P1			P1						P1						✓			
Comp-ACT [14]		P1			P1						P1						✓			
HATO [15]		P1	P1							P1	P1					✓	✓			
RDP [16]		P1			P2	P1		P1, P2			P1					✓	✓			
DWF [17]		P1				P1					P1					✓	✓			
TacDiffusion [18]					P1	P1					P1, P3	P3					✓			
FARM [19]		P1						P1			P1, P3					✓	✓			
ViTacFormer [20]		P1						P1				P1					✓			
3D-ViTac [21]				P1				P1				P1				✓	✓			
UniT [22]		P1						P1				P1			VQR	✓	✓			
FuSe [23]	P1	P1						P1				P1		P1 (Audio)	MCA+AR		✓			
ForceVLA [24]	P1	P1			P1	P1					P1							✓		
Tactile-VLA [25]	P1	P1						P1			P1						✓			
TLA [26]	P1							P1									✓			
FILIC [27]		P1				P1						P1, P3					✓			
ManipForce [28]		P1			P1												✓			
OGRL-VT [29]		P1			P1						P1, P3		P1, P3							SC*
ViTaL [30]	P1	P1, P2							P1, P2								✓			
TACT [31]		P1			P1, P2					P1	P1, P2	P1, P2					✓			
VT-HMD [32]	P1	P1								P1		P1		P1 (TI*)	MAE		✓			
TaF-VLA [33]	P1	P1			P1	P1		P1			P1				VQR		✓			
Force-Policy [34]		P1, P2	P1		P1, P2						P1, P2					✓			✓	
FACTR [35]		P1				P1						P1					✓			
MFMIL [36]		P1						P1			P1									SC*
ForceSight [37]	P1	P1	P1, P2		P1, P2						P1, P2						✓			
CRAFT [38]	P1	P1				P1						P3					✓			
EquiContact [39]	P1	P1			P1, P3						P1, P3					✓				
MoDE-VLA [40]	P1	P1			P1	P1				P1		P1					✓	✓		
ForceVLA2 [41]	P1	P1			P1, P3						P1	P1					✓	✓		
ReTac-ACT [42]		P1						P1			P1	P1			MCA+AR		✓		✓	
TacVLA [43]	P1	P1							P1		P1						✓		✓	
KineDex [44]		P1					P1		P1		P1	P1, P2					✓			
OmniVTA [45]		P1						P1, P2			P1, P2				AR	✓	✓		✓	
OmniVTLA [46]	P1	P1							P1	P1	P1						✓			
ViTaS [47]	P1							P1							MCA+AR		✓			
VLA-Touch [48]	P1	P1, P2					P1, P2	P1			P1,P2,P3						✓			
3DTacDex [49]		P1					P1			P1	P1	P1			MAE		✓			
VTAO-BiManip [50]		P1								P1			P1		MAE		✓			
VTLA [51]	P1	P1						P1									✓			
ARCH [52]		P1			P1						P1		P1			✓	✓			
TouchGuide [53]	P1	P1			P1				P1		P1									DS*

Note. Empty cells denote not reported or not used. P1, P2, and P3 denote different use phases defined in this survey. *TI denotes task indicator. SC denotes simple concatenation, and DS denotes diffusion step.

2.3.4 Potential Modalities

Audio provides acoustic cues about robot-object-environment interactions. The signal can be represented either as a one-dimensional waveform over time or transformed into a spectrogram with temporal and frequency dimensions. Audio captures interaction-related cues, such as collision events, material properties, and task progress during tasks such as liquid pouring, supporting contact-rich manipulation when visual observations are unavailable or insufficient.

Task Indicator provides discrete high-level signals that specify which task a policy should execute. It is commonly used in multitask policy learning to condition a unified policy on different task objectives while maintaining a shared multimodal feature space. For example, VT-HMD [32] trains a unified multitask manipulation policy conditioned on a task-specific one-hot identifier, enabling the same policy to switch among different manipulation tasks.

2.4 Multimodal Representation Learning

Multimodal perception can provide rich complementary sensory information at different stages of task execution. However, effectively utilizing and encoding these modalities through multimodal representation learning is still challenging. Desirable representations should satisfy two requirements: (1) semantically related observations from diverse modalities should be close in a shared latent space, and (2) each representation should preserve sufficient fine-grained, modality-specific information for downstream manipulation tasks [139, 140, 141].

To learn multimodal representations that are both informative and semantically aligned, existing methods often introduce reconstruction- and alignment-based auxiliary objectives. Reconstruction encourages encoded representations to preserve task-relevant modality information by recovering raw observations [4, 22, 32, 33, 42, 46, 47, 49, 50, 130, 142]. In contrast, alignment projects diverse modalities into a shared latent space and pulls together observations associated with the same semantic concept, physical state, or task event [23, 42, 47, 143, 144, 145, 146, 147]. Together, these objectives reduce modality gaps and provide informative representations that allow downstream policies to better exploit complementary multimodal cues.

We group these methods into Auxiliary Reconstruction (AR), Masked Autoencoding (MAE), Vector Quantization (VQ)-based Reconstruction (VQR), and Multimodal Contrastive Alignment (MCA).

2.4.1 Auxiliary Reconstruction (AR)

Auxiliary reconstruction requires encoded features to preserve modality-specific and task-relevant information. Instead of optimizing only a task-specific objective, such as trajectory imitation loss, it asks the model to retain information sufficient to recover the original observation [42, 148, 149, 150]. This is particularly important for fine-grained or task-critical modalities that may be overlooked in end-to-end policy learning. For example, ReTac-ACT [42] reconstructs raw tactile inputs from tactile latent tokens, encouraging the tactile encoder to capture contact geometry and deformation patterns rather than collapsing into generic visual features.

Given the encoded feature of modality m, denoted as $\mathbf{F}^m$, a modality-specific decoder $D_m(\cdot)$ reconstructs the corresponding raw observation $\mathbf{X}^m$. The auxiliary reconstruction objective can be written as

$$\mathcal{L}^m_{\text{rec}} = \|\mathbf{X}^m - D_m(\mathbf{F}^m)\|_2^2, \tag{1}$$

where $\mathbf{X}^m$ is the original observation of modality m, and $D_m(\mathbf{F}^m)$ is its reconstruction from the encoded feature.

2.4.2 Masked Autoencoding (MAE)

Masked Autoencoding (MAE) reconstructs only the missing parts of the input rather than the full observation from encoded features. It first masks a subset of input tokens and then requires the model to infer the masked content from the remaining visible tokens. By solving this incomplete-observation prediction problem, MAE encourages the representation to capture both intra-modal structure and inter-modal complementarity, such as spatial dependencies within each modality and cross-modal correlations between vision and touch [5, 142, 151, 152]. For example, M3L [5] uses multimodal MAE to reconstruct visual pixels and tactile taxels, learning compact vision-tactile representations for reinforcement learning.

For modality m, let Ω_m denote the set of masked token indices, and let $\mathbf{x}_j$ and $\hat{\mathbf{x}}_j$ denote the original and reconstructed token at index j, respectively. The MAE loss is computed only over the masked tokens:

$$\mathcal{L}^m_{\text{MAE}} = \frac{1}{|\Omega_m|} \sum_{j \in \Omega_m} \|\mathbf{x}_j - \hat{\mathbf{x}}_j\|_2^2 . \tag{2}$$

Here, $|\Omega_m|$ is the number of masked tokens. In multimodal settings, the reconstruction can be conditioned on visible tokens from the same modality as well as complementary visible tokens from other modalities. Intuitively, MAE forces the model to infer missing sensory information from context. This makes the learned representation less dependent on isolated local cues and more sensitive to structural regularities and cross-modal relationships.

2.4.3 VQ-based Reconstruction (VQR)

Vector Quantization (VQ)-based reconstruction learns structured multimodal representations by mapping continuous features to a finite set of learnable codebook embeddings. This quantization encourages compact and noise-robust representations, which are valuable for high-dimensional and sensor-dependent modalities such as tactile signals [22, 153, 154, 155]. For example, UniT [22] uses a VQGAN autoencoder to learn tactile representations from GelSight images. By quantizing tactile features into discrete codes, VQ regularization preserves salient contact geometry and force patterns while making the latent space more robust to sensor noise.

Given continuous latent tokens $\mathbf{F}^m = \{\mathbf{f}_j\}_{j=1}^{N_m}$ for modality m, VQ maps each token to its nearest codebook entry:

$$k_j = \arg\min_k \|\mathbf{f}_j - \mathbf{c}_k\|_2^2, \quad \mathbf{q}_j = \mathbf{c}_{k_j}. \tag{3}$$

where $\mathbf{f}_j$ is the j-th continuous latent token, $\mathbf{c}_k$ is the k-th learnable codebook embedding, k_j is the selected code index, and $\mathbf{q}_j$ is the corresponding quantized token. The quantized tokens are then decoded to reconstruct the original sensory input. The VQ objective is introduced to stabilize the discrete latent space:

$$\mathcal{L}_{\text{VQ}}^m = \|\text{sg}[\mathbf{F}^m] - \mathbf{Q}^m\|_2^2 + \beta \|\mathbf{F}^m - \text{sg}[\mathbf{Q}^m]\|_2^2 \tag{4}$$

Here, $\mathbf{Q}^m$ denotes the quantized representation formed by all quantized tokens, $\text{sg}[\cdot]$ is the stop-gradient operation, and β controls the strength of the commitment term. The first term updates the codebook embeddings toward the encoder outputs, while the second term encourages the encoder outputs to stay close to their selected discrete codes. In practice, this VQ objective is combined with a reconstruction loss so that the discrete codes remain both stable and informative.

2.4.4 Multimodal Contrastive Alignment (MCA)

Multimodal contrastive alignment aims to learn a shared latent space for diverse sensory modalities. Instead of reconstructing inputs, it pulls together cross-modal representations associated with the same semantic concept, physical state, or task event while pushing apart mismatched pairs [23, 42, 47, 144, 145, 146, 147]. It is essential in robotics because different sensors often observe the same interaction from complementary perspectives. For example, FuSe [23] applies CLIP-style contrastive learning to align vision, touch, and audio with language instructions, using language as a shared semantic grounding space.

Given two modalities a and b, their encoded features are first pooled and projected into a shared latent space, producing $\mathbf{z}^a$ and $\mathbf{z}^b$. For a minibatch of B paired samples, the matched pair $(\mathbf{z}_i^a, \mathbf{z}_i^b)$ is treated as positive, while mismatched pairs $(\mathbf{z}_i^a, \mathbf{z}_j^b)$ with $j \neq i$ are treated as negatives. The contrastive loss from modality a to modality b is

$$\mathcal{L}_{a\to b} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(\text{sim}(\mathbf{z}_i^a, \mathbf{z}_i^b)/\tau)}{\sum_{j=1}^{B} \exp(\text{sim}(\mathbf{z}_i^a, \mathbf{z}_j^b)/\tau)}, \tag{5}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity and τ is the temperature parameter controlling the sharpness of similarity comparison.

A symmetric objective is commonly used so that both modalities are aligned toward each other:

$$\mathcal{L}_{\text{align}}^{a,b} = \frac{1}{2}(\mathcal{L}_{a\to b} + \mathcal{L}_{b\to a}). \tag{6}$$

Intuitively, MCA teaches different modalities to agree on what they are observing. It does not require each modality to reconstruct raw sensory details, but instead encourages representations from vision, touch, audio, force, or language to meet in a common semantic space when they describe the same interaction. MCA can also support low-resource modality adaptation, allowing a newly introduced modality to benefit from semantic structures learned from data-rich modalities during pretraining.

2.4.5 Task-Relevant Supervision

Among the surveyed methods that explicitly introduce representation-learning objectives, reconstruction- and alignment-based objectives are the most common. We also acknowledge that task-relevant supervised signals

can directly inject manipulation semantics into multimodal representations. Such supervision can arise either from joint encoder training with action-generation objectives or from auxiliary pre-training objectives defined by structured task priors.

The most direct source of such supervision is the action-generation loss, which encourages multimodal representations to encode control-relevant physical information for producing precise motor commands. Representative examples include diffusion-denoising objectives in Diffusion Policy [56], flow-matching objectives in π_0 [57], and next-action-token prediction in OpenVLA [156]. Since these losses are tied to the policy architecture itself, we discuss the corresponding action-generation models in detail in Section 3.

Beyond the primary action loss, recent works further introduce auxiliary objectives that provide structured task priors for multimodal representation learning. These supervision signals include gaze regions [157], affordances [158, 159], keypoints [160, 161], masks and poses [162], and multitask objectives [163, 164]. Compared with raw sensory reconstruction, these targets are more directly related to manipulation; compared with full action trajectories, they are often easier to annotate, predict, or optimize. As a result, they can serve as interpretable intermediate variables for grounding multimodal representations. For example, ReconVLA predicts gaze regions of manipulated objects to promote target-centric visual grounding [157], while MolmoAct predicts depth-aware perception tokens and spatial object traces to enhance 3D action reasoning before low-level control [161].

2.5 Multimodal Fusion

Representation-learning objectives such as reconstruction and contrastive alignment can produce informative modality-specific or shared representations. However, robot policies must further integrate these representations into task-level features for planning and control. This fusion process is non-trivial because different modalities offer distinct and complementary strengths that vary across tasks, execution phases, and environments. For instance, vision often supports global scene understanding, whereas tactile and force signals are crucial during local contact-rich interactions.

Accordingly, effective multimodal fusion should go beyond simple feature aggregation by suppressing redundant information, preserving complementary cues, and regulating modality selection and cross-modal interaction during action generation. Based on these design considerations, we organize existing methods into FiLM injection, attention conditioning, mixture-of-experts, and gated filtering. We then discuss additional potential fusion directions, highlighting possible trends beyond the tactile- and force-centric studies summarized in Table 2.

2.5.1 FiLM Injection

FiLM injection provides a lightweight mechanism for incorporating multimodal information into action generation. Rather than directly concatenating all modality tokens, it summarizes multimodal representations into compact conditioning vectors and uses them to modulate intermediate action features through feature-wise scaling and shifting [56, 165, 166]. This mechanism is particularly useful in diffusion-based robot policies, where noisy action tokens need to be denoised under task- and observation-dependent conditions.

Given noisy action tokens $\mathbf{A}_t$ at diffusion timestep t, FiLM generates a scale parameter $\boldsymbol{\gamma}^m$ and a shift parameter $\boldsymbol{\beta}^m$ from modality m, and modulates the action tokens as

$$\widetilde{\mathbf{A}}_t = \boldsymbol{\gamma}^m \odot \mathbf{A}_t + \boldsymbol{\beta}^m. \tag{7}$$

Here, $\mathbf{A}_t$ denotes the noisy action tokens, $\widetilde{\mathbf{A}}_t$ denotes the modulated action tokens, and $\odot$ denotes element-wise multiplication. The scale and shift parameters are usually produced by a small network from pooled modality features, allowing the sensory context to directly control how action features are amplified, suppressed, or shifted.

2.5.2 Attention Conditioning

Attention conditioning performs token-level multimodal fusion through interactions between action tokens and modality-specific tokens. Unlike global conditioning mechanisms such as FiLM, attention allows action generation to selectively query task-relevant multimodal evidence. This facilitates fine-grained cross-modal

reasoning, such as linking action steps to visual object positions, tactile contact patterns, or force-induced interaction states.

The core operation is scaled dot-product attention:

$$\mathrm{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V}. \tag{8}$$

Here, $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ denote query, key, and value tokens, respectively, and d is the feature dimension used for scaling. In self-attention-based conditioning, action tokens and modality tokens are concatenated into a single sequence, from which $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are jointly computed [55, 57, 58]. In cross-attention conditioning, $\mathbf{Q}$ is computed from the action tokens, whereas $\mathbf{K}$ and $\mathbf{V}$ are computed from the modality tokens, allowing the action representation to retrieve relevant sensory information [54, 56, 167]. Different modalities can also be incorporated through distinct conditioning pathways, such as joint self-attention for image-action interaction, adaptive layer normalization for proprioception, and cross-attention for tactile signals [168].

2.5.3 Mixture-of-Experts (MoE)

Instead of processing all multimodal features with a single shared fusion module, Mixture-of-Experts (MoE) dynamically routes tokens to a set of specialized expert networks. Through sparse routing, different tokens can be processed by different experts that specialize in particular modalities, task phases, or interaction patterns [24, 41, 169, 170, 171]. This property is well suited to robotic manipulation, where the relevance of each modality changes throughout execution. For example, in contact-rich insertion, visual and language tokens may be routed to experts that model object geometry and task semantics during the approach phase, whereas force and tactile tokens may be routed to contact-sensitive experts after contact is detected.

For each modality token $\mathbf{f}_j$, a router predicts expert weights and selects a small subset of experts. The updated token is then computed as a weighted combination of the selected expert outputs:

$$\widetilde{\mathbf{f}}_j = \sum_{i \in \mathcal{S}_j} r_{j,i} E_i(\mathbf{f}_j). \tag{9}$$

Here, $\mathcal{S}_j$ denotes the set of selected experts for token $\mathbf{f}_j$, $E_i(\cdot)$ is the i-th expert network, and $r_{j,i}$ is the routing weight assigned to that expert. The routing weights are typically produced by a softmax router, while only the top-ranked experts are activated for computational efficiency. MoE does not directly compute pairwise interactions between action tokens and modality tokens. Instead, it adaptively transforms modality features before they are injected into the policy through concatenation, FiLM, or attention.

2.5.4 Gated Filtering

Gated filtering dynamically controls each modality's contribution to multimodal fusion according to the current task phase or interaction state. Rather than continuously injecting all multimodal features, it assigns state-dependent gates to modality representations, allowing the policy to emphasize useful information and suppress redundant or noisy information [11, 34, 42, 43, 45, 172]. This mechanism is particularly suitable for phase-dependent modalities. For example, tactile signals are informative after contact is established but may be uninformative or noisy during free-space motion. By filtering modality features before fusion, gating reduces cross-modal interference and encourages task-relevant sensory integration.

Given modality features $\mathbf{F}^m$ and a task-dependent state $\mathbf{s}_t$, gated filtering can be written as

$$\widetilde{\mathbf{F}}^m = \sigma(G_m(\mathbf{s}_t)) \odot \mathbf{F}^m. \tag{10}$$

Here, $G_m(\cdot)$ is a gating network for modality m, $\sigma(\cdot)$ is the sigmoid function that produces gate values between 0 and 1, and $\mathbf{s}_t$ denotes the current task-dependent state, such as the end-effector pose, contact state, or execution phase. The resulting $\widetilde{\mathbf{F}}^m$ is the filtered modality representation, which can then be fused into action tokens through concatenation, FiLM, or attention-based conditioning.

Intuitively, gated filtering works like a learned sensory switch. It does not assume that every modality is always useful; instead, it lets the policy decide when to attend to each modality and when to suppress it, which is especially important for contact-rich manipulation with phase-dependent sensory relevance.

2.6 Discussions on Potential Directions

Denoising-Stage Modality Composition performs multimodal fusion at the denoising or sampling level for diffusion- or flow-matching-based robot policies. Instead of fusing different modality features into a single representation before action generation, it composes modality-specific guidance during the iterative denoising process [52, 173]. This design is motivated by the observation that different modalities may provide complementary constraints at different sampling stages. For example, visual observations can guide coarse trajectory generation in early denoising steps, whereas tactile or force feedback can refine contact-sensitive constraints in later steps.

Let $\mathbf{A}_t^\tau$ denote the noisy action representation at denoising step τ, and $\mathbf{F}^a$, $\mathbf{F}^b$ represent two modality features. During early denoising, the policy follows modality-a guidance:

$$\hat{\boldsymbol{\epsilon}}^\tau = \boldsymbol{\epsilon}_\theta(\mathbf{A}_t^\tau, \tau, \mathbf{F}^a), \quad \tau \in \mathcal{T}_{\text{early}}, \tag{11}$$

$$\mathbf{A}_t^{\tau-1} = \text{Denoise}(\mathbf{A}_t^\tau, \hat{\boldsymbol{\epsilon}}^\tau, \tau). \tag{12}$$

During later denoising, a modality-b guidance model S_ϕ^b further provides a task-relevant guidance score and the denoising direction is then modified by the score gradient:

$$s_\phi^b = S_\phi^b(\mathbf{A}_t^\tau, \mathbf{F}^a, \mathbf{F}^b). \tag{13}$$

$$\hat{\boldsymbol{\epsilon}}^\tau = \boldsymbol{\epsilon}_\theta(\mathbf{A}_t^\tau, \tau, \mathbf{F}^a) - \eta_\tau \nabla_{\mathbf{A}_t^\tau} s_\phi^b, \quad \tau \in \mathcal{T}_{\text{late}}, \tag{14}$$

where η_τ is the guidance scale. The guided direction is then used to update $\mathbf{A}_t^\tau$ toward the final action sequence $\mathbf{A}_t^0$.

Balanced Multimodal Fusion aims to mitigate modality dominance during multimodal policy learning [174, 175]. In multimodal optimization, highly predictive modalities such as vision or language can dominate gradient updates, causing weaker but task-critical modalities, such as tactile or force, to be insufficiently explored. This imbalance reduces the policy's ability to exploit complementary multimodal information, especially in contact-rich manipulation, where non-visual modalities may only be informative during specific interaction phases.

A representative strategy is to dynamically adapt the learning losses according to each modality's contribution. Instead of assigning the same optimization strength to all modality branches, balanced learning estimates modality dominance and adjusts gradient updates to suppress over-dominant modalities or enhance under-utilized ones:

$$\nabla_{\theta_m} \mathcal{L} \leftarrow \rho^m \nabla_{\theta_m} \mathcal{L}, \quad \rho^m = \Phi\left(s^1, s^2, \ldots, s^M\right). \tag{15}$$

Here, θ_m denotes the parameters of modality branch m, s^m denotes the estimated contribution or dominance score of modality m, and ρ^m is the corresponding gradient modulation coefficient produced by $\Phi(\cdot)$. In practice, ρ^m can reduce the update strength of over-dominant modalities or increase the update strength of under-optimized modalities.

Foresight Modality Imagination refers to predicting future multimodal observations or latent dynamics from historical multimodal features through a **world model** [176]. Unlike directly fusing only current observations, this strategy imagines how the physical interaction may evolve in the near future, such as whether slippage or collision may occur, or how tactile deformation may change. As a result, it can guide and refine action generation before actual collision or object deformation happens, thereby reducing the risk of unsafe or damaging interactions. For example, OmniVTA employs a visuo-tactile world model [45] to predict future latent tactile representations. The predicted tactile foresight is then passed to a reflexive tactile controller, which refines coarse actions by correcting deviations between predicted and observed tactile signals. In this way, the policy not only reacts to currently observed contact signals but also anticipates future contact evolution before it is fully observed.

Given historical multimodal features, a world model predicts future observations or latent states:

$$\mathbf{X}_{t+1:t+H}^{\text{fs}} = W_\theta\left(\{\mathbf{F}_{t-H:t}^m\}_{m=1}^M\right), \tag{16}$$

where H denotes the horizon length. The predicted foresight signal is then encoded into a foresight representation $\mathbf{F}_t^{\text{fs}}$, which conditions the downstream policy to guide or refine action generation.

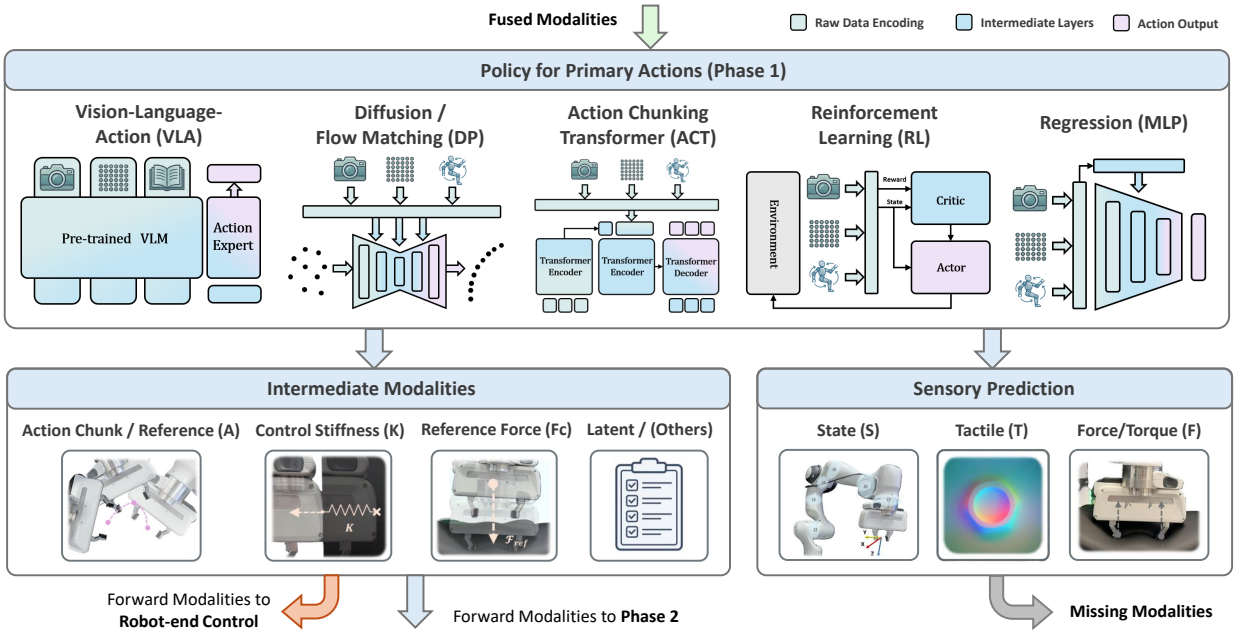

Figure 4 Overview of the Phase-1 primary action-generation policy, explicitly showing the input-output relationships and the common model architectures used in the surveyed papers.

3 Phase 1: Primary Action Generation

The primary action policy, referred to here as the Phase-1 policy, serves as the core decision-making module in the proposed framework. Except for methods that explicitly address low-level control, most robot learning approaches rely on such a primary policy to map encoded and fused multimodal observation embeddings, which may be difficult to interpret intuitively, into robot actions or parameters with explicit physical meaning.

On the input side, the primary policy aggregates encoded features from Section 2. Depending on the available sensors and target task, the policy may take raw sensory inputs, fused multimodal representations, or task-relevant features that facilitate action generation. On the output side, it predicts motion trajectories for direct robot execution or generates coarse trajectories that can be further refined by a second-phase policy, as discussed in Section 4. Alternatively, the primary policy may produce outputs that guide downstream controllers, such as contact-force references and interaction-related control stiffness.

Figure 4 provides an overview of the primary action policy, including abstract model architecture visualizations, representative input and output examples, and sensory prediction examples. Table 3 summarizes the surveyed papers according to the following entries: model architecture, action-space classification, intermediate modalities predicted in addition to actions, and missing-observation prediction.

3.1 Architecture of Primary Policy

In this survey, we categorize the major action-generation approaches in the surveyed papers into five classes: regression-based policies, reinforcement learning (RL), diffusion/flow-matching policy (DP), action chunking transformer (ACT), and vision-language-action (VLA) models.

3.1.1 Regression-based Policies

Regression-based policies formulate robotic action prediction as a supervised learning problem. Rather than modeling complex action distributions, they learn a deterministic mapping from sensory inputs to continuous control commands. This architecture typically uses feature-extraction backbones paired with an action prediction head to generate actions from multimodal observations. Given a multimodal observation feature o_t at time step t, the policy π predicts an action $\hat{a}_t$. The network is trained by minimizing the discrepancy

between the predicted action and the ground-truth expert demonstration a_t, commonly using the mean squared error (MSE) loss:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \|\pi(o_{t,i}) - a_{t,i}\|_2^2 \tag{17}$$

Here, N denotes the number of samples in the batch. This objective encourages the policy to reproduce expert actions accurately, providing a simple and effective policy learning baseline for direct position or force control [32, 36, 37].

While regression-based policies benefit from architectural simplicity and high inference frequency suitable for real-time control, they struggle with multimodal action distributions. When multiple valid behaviors exist for the same observation, the model tends to average them, producing hesitant or physically infeasible actions.

3.1.2 Reinforcement Learning (RL)

RL learns optimal control strategies through trial-and-error interactions rather than relying solely on expert demonstrations. By formulating robot action prediction as a Markov Decision Process (MDP), RL can learn complex and contact-rich behaviors under highly nonlinear dynamics [177, 178, 179, 180, 181, 182, 183]. Many modern RL methods for robotics adopt an actor-critic architecture, where a policy network, or actor, maps multimodal observations to actions, and a value network, or critic, estimates cumulative returns. To maximize the expected long-term return under an environment reward R_t, RL methods often use Proximal Policy Optimization (PPO) [184], which updates the policy by maximizing a clipped surrogate objective:

$$J_{\text{PPO}}(\theta) = \hat{\mathbb{E}}_t \left[\min\left(\rho_t(\theta)\hat{A}_t,\ \text{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right] \tag{18}$$

Here, $\rho_t(\theta)$ denotes the probability ratio of the action under the current policy to that under the previous policy, $\hat{A}_t$ is the estimated advantage function, and ϵ is the clipping threshold.

While RL has strong autonomous exploration capabilities, it suffers from severe sample inefficiency and often requires a large number of interaction steps to converge. Furthermore, its performance depends heavily on carefully engineered reward functions to prevent unintended physical behaviors.

3.1.3 Diffusion/Flow Matching (DP)

DP [56] formulates action generation as a conditional progressive denoising process [185, 186, 187, 188, 189, 190].[1] Unlike deterministic regression, DP transforms Gaussian noise into valid action trajectories through iterative refinement, allowing it to naturally model multimodal action distributions. This helps avoid mode averaging when multiple valid expert behaviors exist for the same sensory context. Given an observation prefix o_t, the policy initializes an action sequence as pure Gaussian noise ϵ. A denoising network, typically implemented using a Transformer or time-conditional U-Net, then iteratively refines this noise. The network is trained using either a score-matching objective (for diffusion) or a vector-field matching objective (for flow matching). For instance, the standard diffusion noise-prediction loss is formulated as:

$$\mathcal{L}_{\text{DP}} = \mathbb{E}\left[\|\epsilon_\theta(x_\tau, \tau, o_t) - \epsilon\|^2 \right] \tag{19}$$

Here, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is the ground-truth random noise vector, x_τ is the noisy action sequence at diffusion timestep τ, o_t denotes the multimodal observation context, and $\epsilon_\theta(\cdot)$ is the noise predicted by the denoising network. While generative denoising policies can generate trajectories that are robust to noisy demonstrations, the iterative denoising process introduces considerable inference latency. This computational cost makes direct deployment in high-frequency closed-loop control challenging.

[1] We use DP to denote both the specific Diffusion Policy and the broader class of generative denoising policies, including flow matching. We acknowledge this overload; despite flow matching relying on vector-field regression instead of standard diffusion, both share the same progressive trajectory generation paradigm.

3.1.4 Action Chunking Transformer (ACT)

ACT [191] formulates action generation as a simultaneous sequence-generation problem. By predicting an action chunk in a single forward pass, ACT enforces temporal consistency and reduces the execution drift commonly observed in step-by-step autoregressive methods. The architecture typically pairs a Transformer encoder-decoder with a Conditional Variational Autoencoder (CVAE). During execution, the decoder uses a sampled latent variable z together with the current observation o_t to generate a continuous trajectory over a time horizon H. The model is trained by minimizing the negative evidence lower bound (ELBO), balancing reconstruction accuracy with latent regularization:

$$\mathcal{L}_{\text{ACT}} = \sum_{h=0}^{H} \|\hat{a}_{t+h} - a_{t+h}\|_1 + \beta \mathcal{D}_{\text{KL}} \left(\mathcal{Q}(z|a_{t:t+H}, o_t) \parallel \mathcal{P}(z) \right) \tag{20}$$

Here, a_{t+h} represents the ground-truth action, $\mathcal{Q}$ is the CVAE posterior encoding the expert action chunk conditioned on the current observation, and $\mathcal{P}(z)$ is the standard normal prior. During inference, z can be set to zero (the mean of the prior) for deterministic action generation. $\mathcal{D}_{\text{KL}}$ denotes the Kullback–Leibler divergence, and β is a scaling hyperparameter [192, 193, 194, 195, 196, 197].

3.1.5 Vision-Language-Action (VLA)

The VLA model unifies semantic reasoning and low-level execution by grounding representations learned by large vision-language models (VLMs) into downstream action generation. This architecture can process multi-view images, tokenized language instructions, and proprioceptive states through a pre-trained VLM backbone. While some VLA formulations, such as RT-1 [54], RT-2 [198], and OpenVLA [156], rely on discretizing physical space to predict autoregressive action tokens, other architectures, such as π_0 [57], ForceVLA, and Tactile-VLA, decode the VLM's rich semantic embeddings directly into continuous robot actions through regression heads, diffusion generators, or action chunking modules. Because the VLA paradigm serves as a multimodal foundation rather than a fixed policy architecture, its training objective depends on the choice of action decoder. Consequently, the optimization objective varies with the generation mechanism: autoregressive cross-entropy (CE) for discrete action-token prediction, standard distance metrics such as L_1 or L_2 loss for continuous control, flow-matching or score-matching losses for diffusion-based generative modeling, and chunk-level reconstruction objectives for trajectory forecasting [199, 200, 201, 202, 203].

VLA models provide strong open-vocabulary understanding and can improve zero-shot generalization, enabling robots to execute unconstrained instructions on unseen objects. However, the large parameter scale of VLM backbones causes substantial training and inference cost. Effectively bridging high-level semantic reasoning with the high-frequency, low-latency continuous control required for precision contact tasks remains a central architectural challenge.

3.2 Output of Primary Policy

After establishing the architecture of the primary policy, we next examine the policy outputs and their formats. These outputs, referred to here as *intermediate modalities*, can generally be categorized into four groups: actions, force references, stiffness-control parameters, and auxiliary predictions.

3.2.1 Actions

Action outputs constitute the most fundamental category of intermediate modalities, as they define the robot's motion behavior in both spatial and temporal dimensions. Depending on task requirements and hardware embodiment, the action space can be formulated in several ways.

End-effector (EEF)-based actions specify the Cartesian pose or displacement of the manipulator in task space. This formulation is suitable when human interpretability is important. Joint-space actions directly specify joint positions, velocities, or torque commands. They are often used in high-degree-of-freedom dexterous systems, such as robotic hands. Direct prediction in joint space can also avoid potential issues introduced by inverse kinematics solvers, such as singularities in redundant manipulators. Delta actions represent relative changes between the current and target states, such as ΔX, ΔY, and ΔZ, rather than absolute target states. They

Table 3 Summary of Phase 1 model architecture choices, action spaces, action horizons, and auxiliary action or prediction outputs.

Method	Architecture	Action (A)					Control Stiffness (K)	Reference Force (Fc)	Other	Sensory Prediction
		Action Space			Action Horizon					
		Delta	EEF	Joint	Per-step	Chunk				
TA-VLA [4]	VLA			✓		✓				Force
FuSe [23]	VLA	✓	✓			✓				
ForceVLA [24]	VLA		✓			✓				
ForceVLA2 [41]	VLA	✓	✓			✓		✓	transition indicator	
Tactile-VLA [25]	VLA		✓			✓		✓		
TLA [26]	VLA	✓				✓				
CRAFT [38]	VLA	✓	✓	✓		✓				
TaF-VLA [33]	VLA		✓			✓				
OmniVTLA [46]	VLA		✓	✓		✓				
VLA-Touch [48]	VLA		✓			✓				
VTLA [51]	VLA		✓		✓					
TacVLA [43]	VLA	✓	✓	✓		✓				
MoDE-VLA [40]	VLA, RL	✓		✓		✓				State
FORGE [3]	RL		✓		✓				early termination action	
M3L [5]	RL	✓	✓		✓					
HIL-SERL [8]	RL		✓		✓			✓		
OGRL-VT [29]	RL		✓		✓					State
VTAO-BiManip [50]	RL			✓	✓					
ViTaS [47]	RL, DP	✓	✓	✓	✓	✓				
ARCH [52]	RL, MP[1]		✓		✓	✓				
VT-HMD [32]	RL, Regression			✓	✓					
UPPFC [1]	DP			✓	✓			✓		Force*
ACP [6]	DP		✓			✓	✓			
AdmitDiff [7]	DP		✓			✓		✓		
DexForce [10]	DP		✓			✓				
FoAR [11]	DP		✓			✓				
ForceMimic [12]	DP			✓		✓		✓		
DIPCOM [13]	DP		✓			✓	✓			
HATO [15]	DP			✓		✓				
RDP [16]	DP								latent action chunk	
DWF [17]	DP	✓				✓				
TacDiffusion [18]	DP		✓		✓			✓		
FARM [19]	DP		✓			✓		✓		
3D-ViTac [21]	DP			✓		✓				
UniT [22]	DP			✓		✓				
ManipForce [28]	DP	✓	✓			✓				
Force-Policy [34]	DP		✓			✓			global feature	
KineDex [44]	DP			✓		✓		✓		
OmniVTA [45]	DP	✓				✓			future tactile signals	
3DTacDex [49]	DP		✓	✓		✓				
TouchGuide [53]	DP			✓		✓				
Comp-ACT [14]	ACT		✓			✓	✓			
Bi-ACT [9]	ACT			✓		✓		✓		
ViTacFormer [20]	ACT		✓	✓		✓				Tactile
FILIC [27]	ACT		✓			✓				Force
ViTaL [30]	ACT		✓			✓				
TACT [31]	ACT			✓		✓				
FACTR [35]	ACT			✓		✓				
EquiContact [39]	ACT		✓		✓		✓			
ReTac-ACT [42]	ACT			✓		✓				
MFMIL [36]	Regression	✓	✓		✓				mode switching signal	
ForceSight [37]	Regression		✓		✓			✓	affordance map, depth map	
FCLM [2]	-									Force*, State*

* The corresponding modality observation occurs in Phase 2 of the architecture; it is repeated here for completeness.

[1] Motion Planning.

are commonly adopted in cross-embodiment learning settings because of their stronger generalization ability and reduced dependence on absolute initial poses. Delta actions are not a parallel category to EEF-space or joint-space actions. Instead, they indicate whether the action is represented in a relative or absolute form.

In addition to spatial representation, action outputs also differ in temporal horizon. Per-step policies generate only the next control action at each control cycle, as commonly seen in early regression-based imitation learning and reinforcement learning architectures. This formulation is suitable for reactive control, as it allows the policy to update its decision at every control step using the latest observations. In contrast, action-chunk-based policies predict a sequence of future actions in a single forward pass, as widely adopted

in Transformer- and diffusion-based architectures such as ACT, Diffusion Policy, and VLA. By modeling temporal correlations across multiple future steps, action chunks can produce smoother and more coherent trajectories [204]. They also provide flexibility during execution: later actions in the predicted chunk can be truncated and replaced by newly predicted actions from subsequent forward passes, allowing the policy to combine short-horizon reactivity with longer-horizon temporal consistency [205, 206].

3.2.2 Reference Force

Beyond motion generation, recent tactile- and force-aware policies sometimes explicitly predict reference force signals to enable safe and stable physical interaction [1, 7, 8, 9, 12, 18, 19, 25, 37, 41, 44]. These reference signals, ranging from Cartesian wrenches to grasping forces, typically serve as key inputs to robot-end force control, where they help produce compliant and force-reactive behavior. In practice, they can be used through different mechanisms, such as feed-forward compensation to offset interaction loads and reduce response latency, task-driven regulation for dynamically modulating grasp intensity, and internal force guidance for balancing motion accuracy with physical safety. Predicting reference force is most meaningful when paired with explicit low-level force control. Such combinations are commonly used in controllers such as admittance control, impedance control, and hybrid force-position control, as discussed in Section 5.

3.2.3 Control Stiffness

Another critical output of the primary policy that naturally works with low-level control is the control stiffness K [6, 13, 14, 39]. The predicted stiffness gains, such as translational and rotational stiffness parameters (K_p, K_r), are typically integrated into impedance or admittance control frameworks to shape robot behavior as a virtual spring-damper system. This modulation allows the robot to balance safety and task performance by permitting intentional trajectory deviations when large contact forces arise in precision tasks. The magnitude of the gain K typically determines the compliant behavior of the robot: a larger gain leads to a more rigid response under contact, while a smaller gain makes the robot more compliant.

3.2.4 Other Auxiliary Modalities

In addition to the core outputs above, auxiliary modalities can also be generated and integrated for system-level coordination and reasoning [3, 16, 34, 36, 37, 41, 45]. Examples include skill- or mode-switching triggers that activate specialized expert modules for highly dexterous subtasks [36], task-progress indicators that estimate completion status and support automatic state transitions [41], affordance maps that highlight target object regions [37], and future latent predictions following the world-model convention [45]. Through these multidimensional outputs, the primary policy connects semantic perception, physical reasoning, and low-level execution, enabling robots to achieve robust interaction in diverse and complex scenarios.

3.3 Sensory Prediction

The primary policy in many recent tactile- and force-aware robotic frameworks also performs sensory prediction, where the model estimates or forecasts sensory observations from one or multiple other modalities. The primary motivation behind this mechanism is to enhance the policy's understanding of complex physical interactions through anticipatory perception, while also compensating for missing, delayed, or unavailable sensory inputs during inference.

Existing works typically perform sensory prediction along three major dimensions: force prediction, state prediction, and tactile prediction. Force prediction aims to estimate interaction forces during manipulation, particularly in scenarios where direct force sensing is unavailable or insufficient. Existing approaches generally include predicting force-related observations through cross-modal learning strategies [2, 4] and treating force prediction as an auxiliary self-supervised objective during policy learning [1]. State prediction focuses on estimating system states, hidden physical properties, or task-progression signals that are difficult to measure directly in the physical setup. Existing approaches generally include estimating privileged physical information from accessible observations [29, 40] and predicting internal robot states, including velocity or orientation, for improved motion stability [2]. Tactile prediction aims to provide the robot with future tactile states from historical sensory sequences through temporal prediction models [20].

Instead of jointly training the observation-prediction module with the action-generation model, another option, especially for force prediction, is to construct a separate observation estimator before policy-model training. This can include analytical estimation of external contact forces based on robot dynamics [27], following ideas from the contact-estimation literature [122, 123]. It can also include learning-based approaches that train dedicated temporal models, such as LSTMs, for torque estimation [207], as studied in learning-based dynamics modeling [124, 125, 126]. NeuralActuator [127] provides a representative example of a separately trained observation estimator for low-cost, servo-driven manipulators. It maps histories of commanded poses, proprioceptive states, tracking errors, and actuator-side telemetry to a torque surrogate using a temporal Transformer.

3.4 Challenges of Single-Phase Policies

A key challenge for the primary policy is to regulate the relative contributions of diverse modalities. Visual and language observations often provide high-level semantic information. When policies target semantic reasoning and long-horizon decision making, vision-language inputs tend to play a dominant role, and a relatively low inference frequency is therefore acceptable. This design is commonly observed in architectures that jointly perform semantic understanding and action generation, such as VLA models.

However, excessive dependence on high-dimensional visual or language encoding can introduce latency and reduce responsiveness, which is particularly problematic when stable contact must be maintained. Accordingly, policies designed for contact-intensive manipulation often benefit from reduced reliance on vision or language and stronger grounding in real-time interaction modalities, such as force feedback. These modalities are typically lower-dimensional and more directly related to contact states, making them better suited for high-frequency motion adjustment and reactive control. This consideration is frequently reflected in more compact and task-specific architectures, such as DP and ACT, where the smaller model scale also makes it more feasible to customize training objectives and introduce intermediate modality prediction, including reference force and control stiffness.

A further strategy is to separate modality usage across multiple models. In such designs, a higher-level model focuses on semantic understanding and low-frequency action generation, while a lower-level model emphasizes force-informed action generation or local action adjustment. This division naturally motivates the multi-phase approaches discussed in the next section, where different stages of the system specialize in distinct levels of perception and decision making.

4 Phase 2: Modality-Discriminative Action Refinement

As discussed in the previous section, the need for both semantic reasoning and fast inference poses challenges for a single-phase model. Using one model to incorporate all available modalities can lead to high inference latency in real-world applications. Thus, it is beneficial to distinguish among different modalities according to their characteristics and utilize them selectively. To this end, multi-phase frameworks address these challenges by decomposing the full action-generation pipeline into multiple stages, where each stage focuses on specific modalities or objectives [55, 208, 209, 210, 211, 212, 213, 214, 215, 216].

In such a multi-phase framework, the first stage, Phase 1, typically learns a primary policy that processes multimodal perception data and generates an initial action plan, as introduced in Section 3. The primary policy mainly focuses on understanding task instructions and generating a coarse action based on measurable sensory information. The second stage, Phase 2, learns a refinement policy or uses a model-based algorithm that takes the output from the primary policy and further refines it using additional or auxiliary modalities, such as force or tactile feedback. The action output of the second stage is then sent to the low-level controller for robot control.

A snapshot of the architecture discussed in this section is shown in Figure 5, including the input-output data flow, candidate model structures, and an illustration of the action-refinement concept. The surveyed papers that follow our hierarchical-architecture criteria are further summarized in Table 4. A more detailed discussion of the purpose of action refinement is provided in Section 4.4.

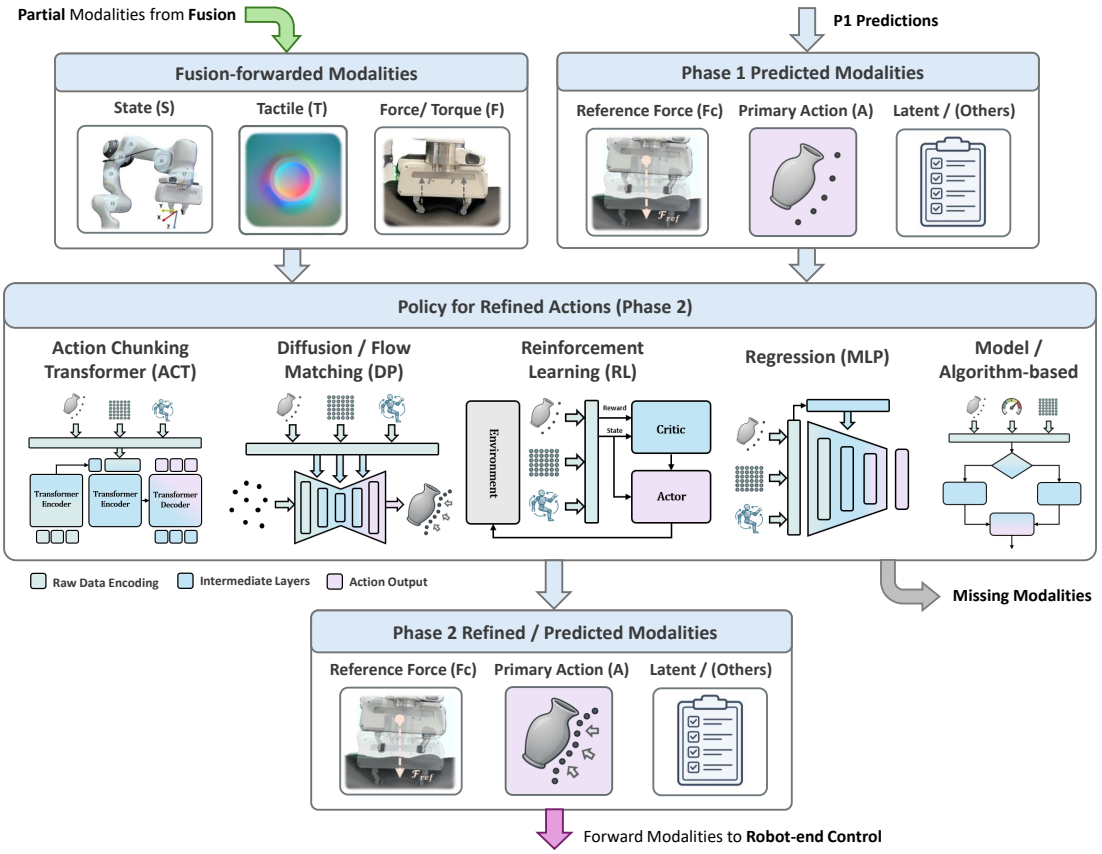

Figure 5 Overview of the Phase-2 action-refinement stage, where forwarded modality fusion and Phase-1 predictions are used for action refinement before low-level control.

4.1 Discriminative Modality Injection

The hierarchical framework enables flexible modality allocation across phases. This section discusses this property by first recapping the major input modalities used in Phase 1 for comparison, followed by the two data streams injected into Phase 2: forwarded input fusion and intermediate predictions from Phase 1.

When supplemented by a Phase 2 policy, the Phase 1 policy typically generates coarse actions primarily driven by vision, language, and state information. These inputs are sometimes augmented with auxiliary modalities, such as tactile sensing for fine-grained contact interpretation, as in ViTaL [30], or force/torque sensing for direct interaction feedback, as in FoAR [11] and RDP [16]. Notable exceptions arise from task-specific demands. For high-precision assembly, TacDiffusion [18] completely omits vision in favor of purely force-proprioceptive feedback. Alternatively, motion-control-centric methods such as FCLM [2] bypass a learned Phase 1 entirely and derive upstream commands directly from human operators.

At Phase 2, the hierarchical framework allows some raw sensory modalities to bypass Phase 1 and feed directly into the refinement stage for real-time action adjustment. This survey refers to this mechanism as *discriminative modality injection*. Four modality types (tactile, force/torque, proprioception, and localized vision) and fusion-phase-inferred features are forwarded in existing works. These inputs appear at different frequencies, as shown in Table 4. Proprioception is the most widely forwarded modality, since Phase 2 requires the current joint or end-effector state to represent the robot state. The other physical modalities are forwarded at different prevalence across methods: force/torque for high-frequency contact feedback, as in FoAR [11] and ForceSight [37]; tactile sensing for contact interpretation, as in TACT [31] and OmniVTA [45]; and vision in the form of wrist-camera observations or local views, as in ViTaL [30] and VLA-Touch [48]. Fusion-phase-inferred modalities include the contact probability predicted in FoAR [11] and the visuo-tactile latents predicted by the world model in OmniVTA [45].

Intermediate predictions produced by Phase 1 can be grouped into three broad categories: action chunks, force commands, and other learned intermediates. Passing an action chunk is the most common choice, with most methods taking the Phase-1 generated action sequence as the refinement target. Force commands are key inputs for methods such as TacDiffusion [18], which refines feed-forward force; KineDex [44], which converts force commands into virtual displacement; and ForceSight [37], which uses a proportional control law with geometric goals. Other intermediate predictions are relatively diverse: RDP [16] produces a latent chunk for decoding; Force-Policy [34] forwards global visual features for fusion with local ones; and ForceSight [37] constructs an affordance map and depth map as inputs to the low-level controller. FCLM [2] again presents a special case, where Phase 2 receives teleoperation commands from a human operator rather than learned policy outputs, although these commands still contain both force and action information.

4.2 Policy Architecture

Similar to the Phase 1 policy, the model architectures used in Phase 2 can be broadly divided into five categories: RL, ACT, DP, regression, and rule/model-based methods that are specific to the refinement stage. While learnable strategies, including RL, ACT, DP, and regression, commonly use the output from Phase 1, their optimization targets differ according to their architectural characteristics. In the following, we briefly discuss how these architectures are adapted for action refinement, rather than focusing on their fundamental concepts and model structures.

The RL architecture trains a policy to compute residual actions over a Phase 1 reference trajectory. For example, UPPFC [1] and FCLM [2] apply PPO to train whole-body force-position controllers for force residuals, whereas ViTaL [30] adds a capped residual from a DDPG policy to the base action.

The ACT architecture appears in Phase 2 as a lightweight variant adapted for high-frequency processing. Instead of using a full transformer, RDP [16] employs an asymmetric tokenizer based on CNN/GRU modules to decode latent chunks at 20–30 Hz.

The DP architecture uses stochastic diffusion to refine action plans rather than generating them from scratch. Force-Policy [34] denoises local motions by utilizing features inherited from the global policy, while VLA-Touch [48] diffuses a VLA output trajectory toward fine-grained expert distributions.

Focusing on latency, the regression architecture employs shallow networks supervised by MSE or L1 losses to predict high-frequency reflexive corrections. A representative example is the three-layer MLP in OmniVTA [45], which explicitly targets single-step modifications.

Despite the broader robot learning context, rule/model-based architectures are still frequently used because of their irreplaceable physical priors in contact-rich manipulation. These methods implement explicit, data-free control logic: FoAR [11] cascades contact-switching state machines; TacDiffusion [18] employs explicit dynamic-system filters to shape force commands; KineDex [44] adapts to external force using a fixed-stiffness admittance-mimicking mechanism; and TACT [31] and ForceSight [37] use admittance tuning and proportional control, respectively. Collectively, these methods illustrate how learning- and model-based approaches complement each other in action refinement, even in data-driven robot learning pipelines.

4.3 Output of Refinement Policy

The output of Phase 2 is forwarded to Phase 3 for execution and can be broadly categorized into actions, force commands, and other intermediate representations. In most methods, Phase 2 outputs a refined action that is passed to the low-level controller. TacDiffusion [18] is one of the few works whose Phase 2 produces a force command, specifically a refined feed-forward force. Force-Policy [34] presents a more unusual case, where Phase 2 simultaneously outputs a force command, an action, and other representations, such as an interaction frame and a selection mask.

Within the action-output channel, methods can be further distinguished by how the action is expressed. A common pattern is that many Phase 2 outputs are incremental rather than absolute trajectories generated from scratch. ViTaL [30] outputs a residual added to the base action from Phase 1; UPPFC [1] and FCLM [2] output joint-level residuals that are composed with a default joint pose to form the execution command; and Force-Policy [34], ForceSight [37], and OmniVTA [45] output incremental or relative actions by utilizing

Table 4 Summary of Phase 2 of multi-hierarchical architectures in force-aware robot learning methods.

Method	Architecture	Visibility of Modalities of Interest[1]					Phase 1 Output	Phase 2 Output Modality						Sens. Pred.
		Force	Tactile	Vision	State	Other		Action Type					Ref Force	
								Ref.[2]	Exe.[3]	Delta	EEF	Joint		
UPPFC [1]	RL	P1		P1	P1,P2		Force, Action		✓		✓			Force
FCLM [2]	RL				P2,P3		Force, Action	✓				✓		Force, State
ViTaL [30]	RL		P1,P2	P1,P2			Action	✓			✓			
Force-Policy [34]	DP	P2,P3		P1,P2	P2		Other	✓			✓		✓	
VLA-Touch [48]	DP	P2	P1	P1,P2	P1,P2,P3		Action	✓			✓			
RDP [16]	ACT-style	P2	P1,P2	P1	P1		Other		✓	✓	✓			
TacDiffusion [18]	Model-Based	P1			P1,P3		Force	✓			✓		✓	
TACT [31]	Model-Based	P2	P1	P1	P1,P2		Action		✓		✓			
ForceSight [37]	Model-Based	P2		P1	P2		Force, Action, Other	✓		✓	✓			
KineDex [44]	Model-Based		P1	P1	P1,P2		Force, Action		✓			✓		
FoAR [11]	Algo-Based	P1,P2		P1	P2	P2	Action		✓		✓			
OmniVTA [45]	Regression		P1,P2	P1	P1,P2	P2	Action		✓	✓	✓			

Note. [1]Similar information is reported in Table 2. This entry is repeated here to indicate whether and how discriminative modality injection is handled in each paper.
[2] 'Ref.' indicates reference action command for low-level controllers.
[3] 'Exe.' indicates robot-executable actions.

Phase 1 trajectory priors. This pattern is consistent with the role of Phase 2 as a refiner, which makes local corrections based on the draft from Phase 1 or a reference pose. By contrast, a smaller group of methods outputs absolute targets: TACT [31] and KineDex [44] produce joint-level position targets, while FoAR [11] produces an end-effector pose target.

4.4 Objectives of Action Refinement

Serving as the middle layer between the high-level learning policy and the low-level controller, the Phase 2 policy focuses on utilizing auxiliary modalities to generate refined actions [217, 218, 219, 220]. Since existing VLA-based methods in Phase 1 rely heavily on visual observations and are often pre-trained on large-scale visual data, directly incorporating additional modalities into these models requires substantial new data and may introduce distribution shifts that degrade the performance of the primary policy. A multi-phase framework can therefore manage the complexity of multimodal perception by allowing the primary policy to focus on visual information, while the refinement policy integrates additional modalities without directly affecting the primary policy.

Meanwhile, the multi-phase framework also enables the robot to refine its actions at a higher control frequency, thereby improving control performance, as discussed in Section 5. Since compliant control methods usually require high control frequencies for stable force tracking, the Phase 1 agent may not be able to output actions at the required rate due to the computational cost of large model scales and multimodal data processing. The refinement policy can therefore benefit from its smaller model scale, taking the output from the primary policy and refining it asynchronously at a higher frequency.

Further, the refinement policy can process exclusive modality information and generate refined actions that are better suited to the current task phase and localized workspace, that is, post-process generated action chunks according to new modality conditions. This can substantially improve the success rate in certain complex tasks. Overall, the multi-phase framework provides a promising approach to addressing mismatches in frequency and information dependency between decision making and motion control, leading to a hybrid system that supports both high-level reasoning and low-level control [221].

5 Phase 3: Robot-End Control

A central low-level objective of force-aware robot learning is to achieve compliant manipulation, in which the robot adapts its motion based on force feedback or directly regulates the force applied to the environment. Compliant controllers therefore serve as the essential connection between learning-based action generation and reliable robot execution, providing a foundation for stable and compliant behavior. This capability is

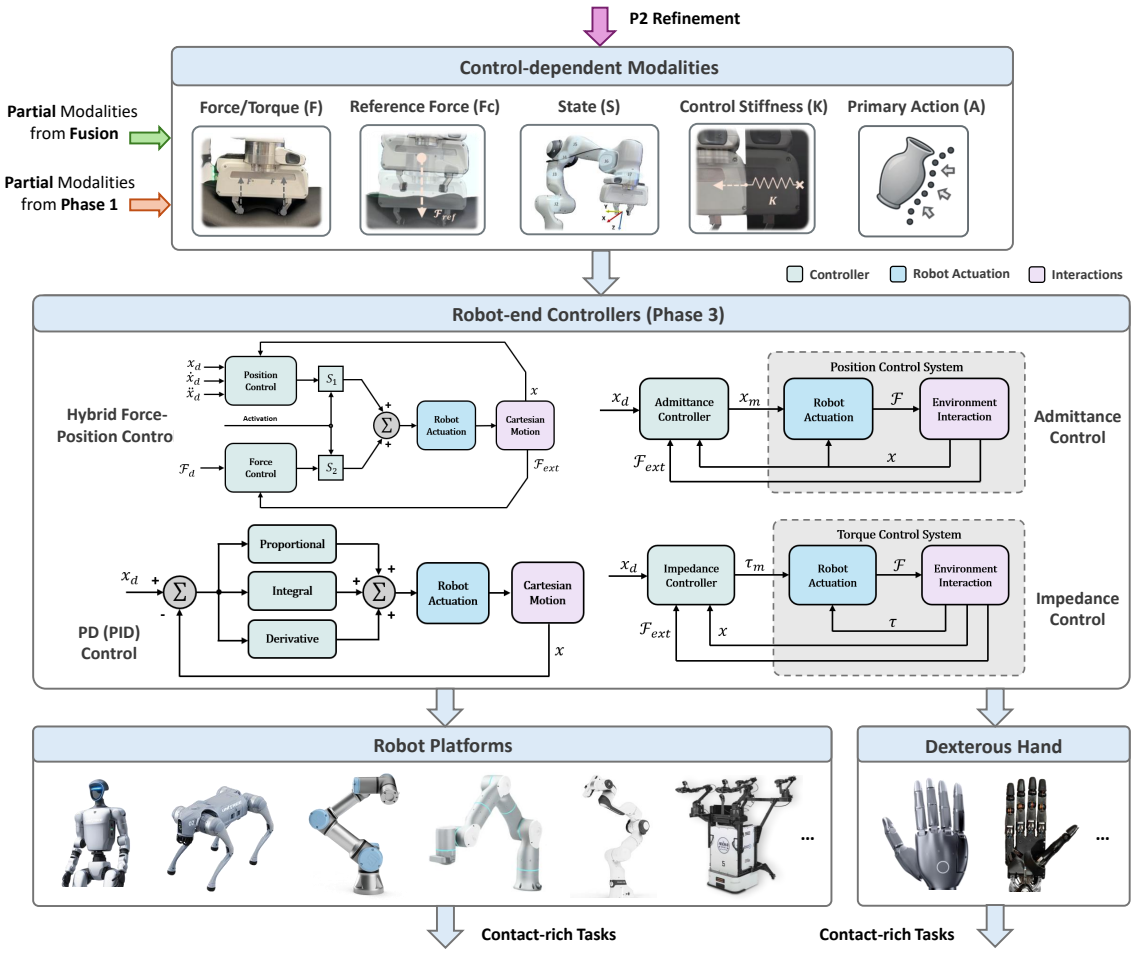

Figure 6 Robot-end control module showing the flow from learned policy outputs and force feedback to compliant low-level control.

particularly important for tasks involving contact-rich manipulation and physical human-robot interaction (pHRI).

In this section, we introduce compliant control methods to explain fundamentally how compliant behavior can be achieved, together with their respective advantages and disadvantages. According to the control objectives and input-output relationships, the major methods can be categorized into PD control, hybrid force-position control, admittance control, and impedance control. Figure 6 illustrates the control-loop block diagrams and multimodal data flow of the robot-end control module. Table 5 summarizes the control-related metrics of the surveyed methods, including controller type, control-demanding modality, robot platform, end-effector type, and available infrastructure-level control modes.

5.1 Preliminaries on Control Theory

5.1.1 Robot Dynamics

Compliant control methods are designed to allow robots to adapt their motion based on interaction forces. A prominent distinction between compliant control and position-based approaches is that the controller attempts to implement a dynamics relation between manipulator variables, e.g., position and velocity, and the force/torque applied to the environment, rather than directly controlling the position or velocity alone [61]. In this process, the manipulator dynamics is one of the key equations in compliant control and can be expressed as [222]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{ext}, \tag{21}$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis and centrifugal matrix, $\mathbf{G}(\mathbf{q})$ is the gravity vector, $\boldsymbol{\tau}$ is the torque control input, $\boldsymbol{\tau}_{ext}$ is the external torque induced by contact, and $\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}$ are the acceleration, velocity, and position vectors, respectively. This equation, formally the inverse dynamics model, describes the joint torque input required by the robot given the robot states $[\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}]$ and a set of inherent robot-specific parameters specified by $[\mathbf{M}(\mathbf{q}), \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}), \mathbf{G}(\mathbf{q})]$. Unlike robot kinematics, the dynamics model builds the relationship between the torque and the motion of the robot, allowing the robot to be controlled through torque to achieve the desired motion.

When making contact with the environment or interacting with humans, the robot will be affected by external forces, as described by the torque term $\boldsymbol{\tau}_{ext}$. In this case, the robot is expected to be compliant and adapt its motion based on the external force [223]. In the Cartesian space, the mass-spring-damper model is usually used to describe the compliant behavior of the robot, which can be expressed as:

$$\mathbf{M}_d(\ddot{\mathbf{x}}_{\text{des}} - \ddot{\mathbf{x}}) + \mathbf{B}_d(\dot{\mathbf{x}}_{\text{des}} - \dot{\mathbf{x}}) + \mathbf{K}_d(\mathbf{x}_{\text{des}} - \mathbf{x}) = \mathbf{F}_{ext}, \tag{22}$$

where $\mathbf{M}_d, \mathbf{B}_d, \mathbf{K}_d$ are the inertia, damping, and stiffness matrices, respectively, $\mathbf{x}_{\text{des}}, \dot{\mathbf{x}}_{\text{des}}, \ddot{\mathbf{x}}_{\text{des}}$ are the desired position, velocity, and acceleration vectors of the end-effector, respectively, $\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}$ are the actual position, velocity, and acceleration vectors of the end-effector, respectively, and $\mathbf{F}_{ext}$ is the external wrench applied to the end-effector. By enforcing this mass-spring-damper relationship, the robot can allow position deviations under external contacts while resisting rapid changes in motion. The corresponding controller will be discussed later in this section.

However, Eqation (22) is expressed in Cartesian space, whereas the manipulator dynamics model is expressed in joint space. To combine these two models, the Jacobian matrix $\mathbf{J}(\mathbf{q})$ is used to transform the wrench from Cartesian space to joint space, which can be expressed as:

$$\boldsymbol{\tau}_{ext} = \mathbf{J}^T(\mathbf{q})\mathbf{F}_{ext}. \tag{23}$$

To this end, the manipulator can be controlled under joint torque command to respond to the external wrench in Cartesian space and adapt its motion accordingly.

5.1.2 PD Control

Position control is one of the aforementioned "position-based" approaches and serves as the most common and fundamental low-level control method. It is mainly based on a proportional-derivative (PD) controller to follow the desired trajectory, which can be expressed as:

$$\boldsymbol{u} = K_p(\mathbf{q}_{\text{des}} - \mathbf{q}) + K_d(\dot{\mathbf{q}}_{\text{des}} - \dot{\mathbf{q}}), \tag{24}$$

where K_p and K_d are the proportional gain and derivative gain, respectively, $\mathbf{q}_{\text{des}}$ and $\dot{\mathbf{q}}_{\text{des}}$ are the desired joint position and velocity, respectively, and $\mathbf{q}$ and $\dot{\mathbf{q}}$ are the actual joint position and velocity, respectively. Here, $\boldsymbol{u}$ denotes the generic control output, which can correspond to a torque command in practice. By adjusting the gains of the PD controller, the robot can follow the desired trajectory through the control output.

However, this method mainly focuses on tracking the desired trajectory and does not provide easily adjustable compliance. Although some existing methods can achieve compliant behavior under position-control mode through force estimation and simulated contact dynamics, traditional compliant control methods are still the most direct and effective way to achieve compliant manipulation.

5.1.3 Impedance Control

Impedance control enforces a desired dynamic relationship between the robot's motion and the forces it experiences, allowing the robot to exhibit compliant behavior by adjusting its impedance parameters. The inputs to an impedance controller are the desired position, velocity, and acceleration of the end-effector, while the output is the joint torque, which can be expressed as [61, 224]:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) - \mathbf{J}^T(\mathbf{q})(\mathbf{M}_d\ddot{\mathbf{e}} + \mathbf{B}_d\dot{\mathbf{e}} + \mathbf{K}_d\mathbf{e}), \tag{25}$$

$$\mathbf{e} = \mathbf{x}_{\text{des}} - \mathbf{x}, \quad \dot{\mathbf{e}} = \dot{\mathbf{x}}_{\text{des}} - \dot{\mathbf{x}}, \quad \ddot{\mathbf{e}} = \ddot{\mathbf{x}}_{\text{des}} - \ddot{\mathbf{x}}. \tag{26}$$

In other words, this formulation specifies a desired trade-off between motion tracking and force tolerance through the impedance parameters $[\mathbf{M}_d, \mathbf{B}_d, \mathbf{K}_d]$. A lower stiffness allows larger positional deviations under external forces, whereas a higher stiffness enforces tighter trajectory tracking.

As illustrated in the impedance-control subfigure of Figure 6, the inner control loop is torque control, while the outer control loop is position control. One advantage of impedance control is that the robot can achieve whole-body compliance without additional force sensors [225]. However, impedance control may be less suitable for tasks that require precise position tracking, because the robot can be affected by external forces or uncompensated dynamics and modeling errors, including joint friction, payload variation, and end-effector weight [226], causing deviations from the desired position. This limitation can be partly mitigated by adjusting the impedance parameters [227, 228, 229, 230, 231] or by combining impedance control with other control methods [232, 233].

5.1.4 Admittance Control

Admittance control, on the other hand, actively describes the relationship between the forces applied to the robot and its resulting motion [63, 234, 235]. The input to an admittance controller is the force applied to the end-effector, while the output is the desired motion command, which can be expressed as [236]:

$$\ddot{\mathbf{x}}_{\text{des}} = \ddot{\mathbf{x}} + \mathbf{M}_d^{-1}(\mathbf{F}_{ext} - \mathbf{B}_d\dot{e} - \mathbf{K}_d e). \tag{27}$$

The desired position can then be obtained by integrating the desired acceleration. When the system moves slowly, the acceleration term can be neglected, and the desired position can be obtained by integrating the desired velocity. Intuitively, admittance control follows the principle that a larger measured external force should induce a larger motion response along the force direction, allowing the robot to yield to the applied force.

Unlike impedance control, the inner control loop of admittance control is position control, while the outer control loop is force control. Admittance control is more suitable for tasks that require precise position tracking, as the controller can actively determine the allowable compromise in position accuracy based on force feedback [237, 238, 239]. However, admittance control may require additional force sensors to accurately measure external forces, and its compliance is limited to the end-effector if no joint-level force/torque information is available.

5.1.5 Hybrid Force/Position Control

Hybrid force/position control combines position and force control to achieve compliant manipulation in selected directions, allowing the robot to maintain a desired position along certain action dimensions while regulating the force applied to the environment along others [60, 240, 241]. The position-control space should be orthogonal to the force-control space, and this decomposition can be defined by a selection matrix $\mathbf{S}$, where $\mathbf{S}$ is a diagonal matrix with binary values indicating the force-control directions [62, 242, 243]. The inputs to a hybrid force/position controller are the desired position and the desired force, while the output can be expressed as:

$$\boldsymbol{u} = \mathcal{F}(\mathbf{S}_2 \cdot e_F) + \mathcal{P}(\mathbf{S}_1 \cdot e), \tag{28}$$

where $\mathbf{S}_1 = \mathbf{I} - \mathbf{S}_2$ is the complementary position-control selection matrix, $\mathcal{F}$ and $\mathcal{P}$ are the force-control and position-control functions, respectively, e_F is the force error between the desired and actual forces, and e is the position error.

This control method enables the robot to achieve both low-stiffness compliance and high-stiffness position control along different directions. It is particularly useful for tasks with clear force-control objectives, such as cutting fruit or wiping a board, where the force-control direction is easy to specify. However, in unstructured or less informative environments, it can be challenging to intuitively disentangle the force-control directions from the position-control directions while maintaining their orthogonality.

5.2 Robot-End Control Implementations

The above introduction provides a basic understanding of compliant control methods and offers a reference for controller selection. Although these controllers serve similar force-compliance purposes, the choice of a

Table 5 Summary of robot-end control configurations in the reviewed tactile- and force-aware robot learning methods.

Method	Controller	Control-Demanding Modality			Robot	EEF type	Control Interface[1]
		Fusion	Phase1	Phase2			
UPPFC [1]	Position			Action	Unitree Z1	Gripper	Position
FCLM [2]	Position	State		Action	Unitree Z1	Gripper	Position
TA-VLA [4]			Action		ALOHA & ROKAE SR	Gripper	Position & Impedance
M3L [5]			Action		Franka	Gripper & Hand	Impedance
ACP [6]	Admittance	Force	Action, K		UR5e	Customized	Position
FORGE [3]	Impedance	State	Action		Franka	Gripper	Impedance
AdmitDiff [7]	Admittance	Force	Action, F_{ref}		UR5-CB3	Hand	Position
HIL-SERL [8]	Impedance	State	Action, F_{ref}		Franka	Gripper	Impedance
DexForce [10]	Impedance	State	Action		– –[3]	Hand	– –
FoAR [11]	Position			Action	Flexiv Rizon	Gripper	Hybrid F/P
DIPCOM [13]	FDCC		Action, K		UR5e	Gripper	Position
Comp-ACT [14]	FDCC		Action, K		UR5e	Gripper	Position
HATO [15]	Position		Action		UR5e	Hand	Position
RDP [16]				Action	Flexiv Rizon	Gripper	Hybrid F/P
TacDiffusion [18]	Impedance	State		F_{ref}	Franka	Gripper	Impedance
FARM [19]	Impedance	State	Action, F_{ref}		Franka	Gripper	Impedance
ViTacFormer [20]			Action		Realman	Hand	Position
3D-ViTac [21]			Action		– –	Gripper	– –
UniT [22]	Position		Action		ALOHA	Gripper	Position
FuSe [23]	Position		Action		WidowX 250	Gripper	Position
ForceVLA [24]			Action		Flexiv Rizon	Gripper	Hybrid F/P
Tactile-VLA [25]	Hybrid F/P	Force	Action, F_{ref}		Unspecified [4]	Gripper	Unspecified
TLA [26]			Action		Unspecified	Gripper	Unspecified
FILIC [27]	Impedance	State	Action		AIRBOT Play	Gripper	Position
ManipForce [28]			Action		Franka	Gripper	Impedance
OGRL-VT [29]	Impedance	State	Action		MELFA	Customized	Position
ViTaL [30]				Action	Ufactory xArm 7	Gripper	Position
TACT [31]	Position			Action	RHP7 Kaleido	Gripper	Position
VT-HMD [32]	Position		Action		– –	Hand	– –
TaF-VLA [33]			Action		Franka	Gripper	Impedance
Force-Policy [34]	Hybrid F/P	Force	Action	Action, F_{ref}	Flexiv Rizon	Gripper	Hybrid F/P
FACTR [35]			Action		Franka	Gripper	Impedance
MFMIL [36]	Impedance	State	Action		Franka	Gripper	Impedance
ForceSight [37]				Action	Stretch RE1	Gripper	Position
CRAFT [38]	Impedance	State	Action		Franka	Gripper	Impedance
EquiContact [39]	Admittance	Force, State	Action, K		Unspecified	Gripper	Unspecified
MoDE-VLA [40]	Position		Action		SharpaNorth	Gripper	Position
ForceVLA2 [41]	Hybrid F/P	Force	Action, F_{ref}		Flexiv Rizon	Gripper	Hybrid F/P
ReTac-ACT [42]			Action		Realman RM75-6FV	Gripper	Position
TacVLA [43]			Action		Franka	Gripper	Impedance
KineDex [44]				Action	Franka	Hand	Impedance
OmniVTA [45]				Action	UFactory xArm7	Gripper	Position
OmniVTLA [46]			Action		UR5	Gripper	Position
ViTaS [47]			Action		Galaxea-R1	Gripper	Position
3DTacDex [49]	Position		Action		JAKA MiniCobo	Hand	Position
VTAO-BiManip [50]	Position		Action		Unitree Z1	Hand	Impedance
VTLA [51]			Action		UR3	Gripper	Position
ARCH [52]				Action	UR10e	Gripper	Position
TouchGuide [53]			Action		BiARX5 & Flexiv Rizon4	Gripper	Position & Hybrid F/P
VLA-Touch [48]	Position & Impedance	State		Action	Franka	Gripper	Impedance
ForceMimic [12]	Position & Hybrid F/P[2]	Force	Action, F_{ref}		Flexiv Rizon	Gripper	Hybrid F/P
Bi-ACT [9]	Bilateral		Action, F_{ref}		OpenMANIPULATOR-X	Gripper	Position
DWF [17]			Action		KUKA IIWA 14 & ABB IRB120	Customized	Position & Impedance

[1] Control Interface denotes the primary infrastructure-level control strategy provided by each robot.
[2] Hybrid F/P indicates the hybrid force/position control approach.
[3] '– –' indicates that the paper focused on hand manipulation and did not cover robot manipulation.
[4] 'Unspecified' indicates that a robot is used in the paper, but the model is not specified.

specific method still depends on the task configuration, such as whether precise position tracking is required, whether additional force sensors are available, and whether the joint-torque command channel is accessible to users. Thus, the controller should be selected carefully according to the physical system settings. Table 5 summarizes the implementations of control methods, together with the control-demanding modalities provided by higher-level learning policies.

Control Mode: The controllers include position control [1, 2, 11, 12, 15, 22, 23, 31, 32, 40, 48, 49, 50], impedance control [3, 8, 10, 18, 19, 27, 29, 36, 38, 48], admittance control [6, 7, 39], hybrid force/position control [12, 25, 34, 41], and Forward Dynamics Compliance Control (FDCC) [13, 14], which are used to control robots using commands from either teleoperation systems or learned policies [244]. Since some methods do

not explicitly report the control method they use, we mark them as unknown to avoid ambiguity. In such cases, however, joint or end-effector position commands are commonly sent directly to the robot.

Control-demanding Modalities: The control-demanding modality refers to the input required by the low-level controller and is categorized into three types according to where the modality is forwarded from: (1) Fusion: This indicates that raw measurements or fused features are used not only by the high-level learning policy but also directly forwarded to the low-level control loop as feedback signals; (2) Phase 1: This indicates that the controller input is forwarded from the primary policy output; and (3) Phase 2: This indicates that the controller input is directly obtained from the refinement policy output. The control-demanding modality reflects how the learning policy supports low-level robot control and how control behavior is coupled with the learning-model output, thereby providing a clearer understanding of modality importance. The corresponding data flow is also illustrated in Figure 2.

Robot Model Selection: To further elaborate on the selection criteria for different platforms, we summarize the robot types and their supported compliant control modes. This information clarifies whether each method leverages the native control interface provided by the robot or employs a custom-designed controller to achieve compliant behavior. The most commonly used robot platform is Franka, which natively provides a stable impedance-control interface. The Flexiv Rizon offers a hybrid force/position-control interface, making it particularly suitable for tasks requiring precise force regulation. The UR5e is equipped with a force/torque (F/T) sensor, enabling admittance control through direct force feedback and facilitating controller design. Other robot platforms primarily provide position-control interfaces, where compliant behavior must be achieved through either custom controller design or action-space manipulation.

End-effector Selection: The end-effector (EEF) type describes how each method interacts with the environment. The most common EEF is the gripper, which is often used to grasp different tools for various tasks. Although a gripper typically has only a single degree of freedom (DoF), it is easy to control and can maintain stable contact. In contrast, a dexterous hand has multiple DoFs and can grasp objects with irregular shapes, but it is also more difficult to control. Therefore, some works focus more on hand control than on the manipulator itself [10, 21, 32]. Apart from grippers and hands, some works use customized tools for specific tasks to achieve better interaction performance [6, 17, 29].

5.3 Practical Considerations for Compliant Control

Robot hardware largely determines the compliant control strategy employed in practice. Many force-aware robot learning methods still use position control as their low-level controller [1, 2, 11, 12, 15, 22, 23, 31, 32, 40, 48, 49, 50], because position control is the most natural user interface and is generally supported by most robot platforms. Under position control, the robot remains rigid and can achieve high positional accuracy, but this rigidity may also lead to unintended contact forces. Such contact forces or collisions may be acceptable for manipulators with low-power motors, but can be dangerous for industrial-level robot manipulators.

Among compliant control methods, impedance control is the most widely used method in the surveyed papers [3, 8, 10, 18, 19, 27, 29, 36, 38, 48]. It can achieve whole-body compliance through feed-forward torque commands without active force sensing. Methods using impedance control are usually based on Franka, KUKA, or ROKAE robot platforms, which provide impedance control as their low-level controller. Meanwhile, methods using hybrid force/position control [12, 25, 34, 41] are usually based on the Flexiv Rizon, which provides a hybrid force/position control interface. These robot platforms provide convenient ways to implement compliant control, allowing learning algorithms to focus mainly on action output rather than learning complex compliant interactions from scratch.

Compliant control provides a solid foundation for force-aware manipulation. However, it also introduces another layer of complexity. If the control gains and parameters are not properly designed and selected, the robot may still experience large contact forces and undesired phenomena such as overshooting. Therefore, the low-level compliant controller should be carefully tested with properly tuned parameters, rather than treated merely as a callable interface.

With reliable robot-end force-control methods, the robot can generally behave more safely by avoiding excessive contact forces while maintaining stable contact for contact-demanding tasks such as peeling and wiping. The

controller and the learning algorithm should complement each other to achieve better compliant interaction performance through appropriate modality exchange, interface connection, and frequency handling.

6 Contact-Rich Manipulation Tasks

Contact-rich manipulation refers to tasks in which the robot perceives and adapts to interaction states, such as contact force, contact location, friction, and object deformation, rather than merely generating contact-free trajectories. Unlike traditional motion-centric manipulation, tactile- and force-aware manipulation requires finer-grained motion generation and explicit contact and collision handling to prevent damage to the robot and its environment. The 53 papers analyzed in this section constitute the TF-ART corpus mapped in Figure 2. Table 6 summarizes the evaluation tasks across these papers, covering 21 manipulation tasks and the tactile, force, and torque sensors used. Figure 7 presents a histogram showing the frequency with which each task appears in the corpus.

6.1 Task Classification

Based on their dominant interaction characteristics, we categorize the surveyed tasks into four groups: precision-contact manipulation, deformable and surface-contact manipulation, dynamic and multi-contact manipulation, and household and long-horizon manipulation. These categories are not strictly disjoint, since many real-world tasks combine geometric constraints, deformable contact, temporal dynamics, and unstructured environments.

These task categories also stress different phases of TF-ART. Precision-contact manipulation mainly emphasizes Phase 2 refinement and Phase 3 compliant control, since small misalignment requires force/tactile-guided correction during contact. Deformable and surface-contact manipulation emphasizes Phase 0 representation learning and Phase 3 force regulation, as material-dependent contact states must be encoded and controlled over extended interactions. Dynamic and multi-contact manipulation stresses the coupling between Phase 2 and Phase 3, where fast refinement and low-latency reactive control are required. Household and long-horizon manipulation places stronger demand on Phase 1 primary policies, since semantic reasoning, object affordance understanding, and multi-stage action generation are needed before local contact adaptation.

6.1.1 Precision Contact Manipulation

This class includes fine-grained tasks, such as peg-in-hole insertion and cap or nut screwing, which require contact-misalignment detection and force-guided motion control. During insertion, the contact region is often occluded and positional errors may be too small to detect visually, so even slight deviations can cause jamming. Cap and handle rotation similarly require coordinated rotational torque and axial force to prevent slipping. These tasks share a common demand for contact localization, fine-grained alignment correction, and compliance control under constrained contact.

6.1.2 Deformable and Surface-contact Manipulation

This class involves deformable objects or irregular surfaces, such as soft or fragile grasping, peeling, and wiping. For soft objects, geometry and dynamics can evolve continuously under contact; thus, the robot must adjust the applied force in real time, balancing the risk of damaging the object against the risk of task failure. Wiping and polishing additionally require stable contact forces over extended trajectories on potentially varying surfaces, which is qualitatively different from the localized contact events in insertion. These tasks share a common demand for compliance- and material-aware manipulation under adaptive force control over extended trajectories.

6.1.3 Dynamic and Multi-contact Manipulation

This class includes time-critical or coordinated tasks, such as reorientation, in-hand manipulation, and bimanual manipulation, which require rapid responses to transient contact. Tasks such as dynamic catching require trajectory estimation and contact response within a narrow time window, placing stringent demands on real-time feedback integration. In-hand manipulation requires simultaneous estimation of multiple contact states and fine-grained force control as the object shifts within the gripper. Bimanual tasks require balanced

Table 6 Summary of tactile/force-aware manipulation tasks and sensing modalities across the surveyed methods.

Method	A Precision Contact					B Deformable & Surface-contact					C Dynamic & Multi-contact						D Household and Long-Horizon					Sensor		
	A1 Insertion	A2 Cap/Nut Screwing	A3 Assembly Plug	A4 Bi. Insertion	A5 Rotate Handle/Box	B1 Peeling	B2 Wiping	B3 Cutting	B4 Slide Object	B5 Soft/Fragile Grasp	C1 Mobile Catch	C2 Sliding	C3 Reorientation	C4 In-hand Manip.	C5 Bi. Lifting	C6 Bi. Wiping	D1 Open/Close Door	D2 Pick-and-place	D3 Pouring	D4 Weight Pulling	D5 Push Object	EEF Wrench	Joint Torque	Tactile Sensor
UPPFC [1]							✓										✓							
FCLM [2]																	✓		✓	✓				
FORGE [3]	✓	✓																				✓		
TA-VLA [4]			✓		✓												✓	✓	✓		✓		✓	
M3L [5]	✓													✓			✓							✓
ACP [6]							✓															✓		
AdmitDiff [7]	✓				✓		✓										✓	✓				✓		
HIL-SERL [8]			✓																					
Bi-ACT [9]																	✓	✓					✓	
DexForce [10]		✓							✓				✓				✓	✓				✓		
FoAR [11]						✓	✓															✓		
ForceMimic [12]						✓																✓		
DIPCOM [13]				✓																		✓		
Comp-ACT [14]	✓						✓									✓						✓		
HATO [15]																			✓					✓
RDP [16]						✓	✓								✓								✓	✓
DWF [17]																							✓	
TacDiffusion [18]	✓																					✓	✓	
FARM [19]	✓									✓														✓
ViTacFormer [20]	✓	✓					✓																	✓
3D-ViTac [21]	✓									✓														✓
UniT [22]	✓									✓														✓
FuSe [23]										✓														✓
ForceVLA [24]	✓					✓	✓															✓	✓	
Tactile-VLA [25]	✓						✓			✓														✓
TLA [26]	✓																							✓
FILIC [27]	✓	✓																						
ManipForce [28]	✓		✓														✓					✓		
OGRL-VT [29]													✓											
ViTaL [30]	✓																							✓
TACT [31]													✓									✓		✓
VT-HMD [32]		✓										✓	✓											✓
TaF-VLA [33]	✓						✓			✓								✓				✓		✓
Force-Policy [34]	✓						✓															✓		
FACTR [35]															✓			✓					✓	
MFMIL [36]																								✓
ForceSight [37]												✓						✓				✓		
CRAFT [38]	✓		✓				✓																✓	
EquiContact [39]	✓						✓											✓				✓		
MoDE-VLA [40]			✓			✓				✓				✓									✓	✓
ForceVLA2 [41]			✓				✓															✓		
ReTac-ACT [42]	✓																							✓
TacVLA [43]	✓	✓	✓															✓						✓
KineDex [44]	✓	✓	✓															✓						✓
OmniVTA [45]			✓			✓	✓	✓		✓			✓											✓
OmniVTLA [46]																		✓						✓
ViTaS [47]	✓		✓				✓				✓			✓	✓			✓						✓
VLA-Touch [48]						✓	✓											✓						✓
3DTacDex [49]			✓										✓				✓							✓
VTAO-BiManip [50]																								✓
VTLA [51]	✓																							✓
ARCH [52]	✓																					✓		
TouchGuide [53]	✓					✓	✓			✓														✓

force regulation and coordinated trajectory generation across two manipulators. These tasks share a common demand for rapid, multi-point coordination of force and tactile feedback under partially observable contact.

6.1.4 Household and Long-horizon Manipulation

This class involves long-horizon, multi-stage tasks, such as pick-and-place and door opening, in unstructured but common daily environments. While pick-and-place is traditionally treated as vision-centric, the tactile-

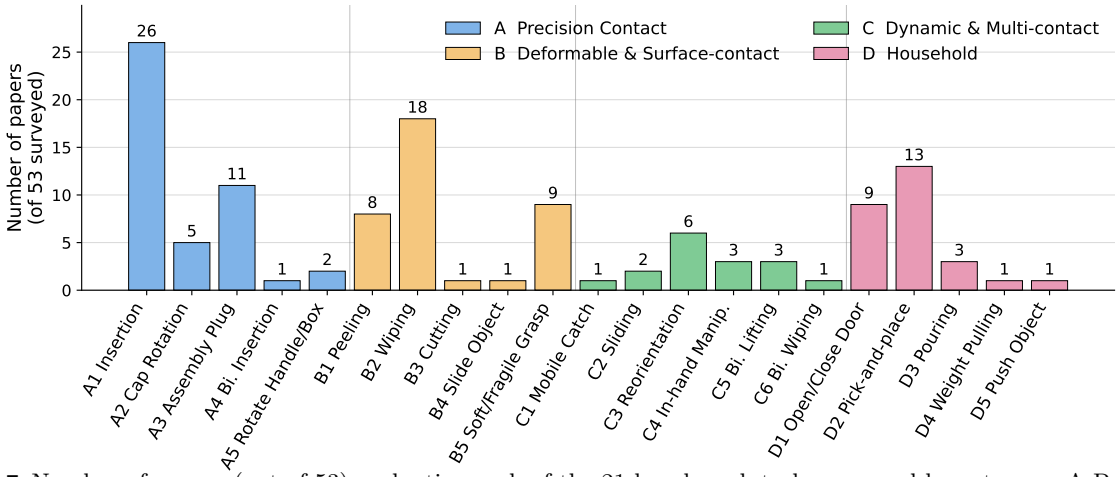

Figure 7 Number of papers (out of 53) evaluating each of the 21 benchmark tasks, grouped by category. A Precision Contact (blue), B Deformable & Surface-contact (orange), C Dynamic & Multi-contact (green), D Household and Long-Horizon Manipulation (pink). The numeric label above each bar is the paper count.

and force-aware variants surveyed here involve fragile objects, uncertain grasp stability, deformable materials, or cluttered contact conditions that require contact feedback during execution. Opening a door or drawer requires constrained contact, rotational motion, and friction-aware control; cooking-related tasks typically integrate grasping, cutting, and container manipulation within a single scene. In this context, material texture and friction can vary widely across daily objects, and visual occlusion is more common in complex household environments, making contact perception a primary requirement. These tasks share a common demand for long-horizon, generalizable force-aware manipulation across diverse and unstructured settings.

6.1.5 Summary of Task Distribution

These four categories capture distinct perspectives on physical interaction intelligence: precision contact emphasizes force-guided interaction under tight geometric constraints; deformable and surface-contact manipulation emphasizes compliance and material awareness over extended trajectories; dynamic and multi-contact manipulation emphasizes rapid feedback and multi-point coordination; and household and long-horizon manipulation emphasizes long-horizon generalization in unstructured environments. Together, they trace a progression from localized force-guided contact alignment to long-horizon, multi-stage interaction reasoning. The actual task distribution, however, is strongly skewed, as illustrated in Figure 7. Peg-in-hole insertion (A1, 26 papers), surface wiping (B2, 18), and pick-and-place (D2, 13) together account for nearly half of all task-method pairings. A second tier, including A3, D1, B5, B1, and C3, covers 6–11 papers each and covers most of the remaining studies. This distribution indicates that the field is concentrated on narrowly scoped precision-contact tasks, whereas dynamic interaction, long-horizon manipulation, and multi-contact coordination remain comparatively under-explored. Overall, the task distribution shows that current studies mainly evaluate contact-refinement problems, while long-horizon reasoning and highly dynamic interaction remain less explored.

6.2 Force and Tactile Sensing Devices

Contact-rich interaction does not always require dedicated force or tactile hardware. However, when contact conditions are uncertain, partially observable, or dynamically changing, vision alone cannot reliably infer the relevant interaction states or support timely adaptation, making tactile or force feedback essential. The experimental setups of the 53 surveyed methods therefore place strong emphasis on contact sensing, as shown in Table 6: 25 employ wrist-mounted force/torque (F/T) or joint torque sensing (18 use wrist F/T sensing and 9 use joint torque sensing, with 2 using both), 27 use tactile sensors, 4 combine a force-domain sensor with a tactile sensor, and 5 either estimate force from proprioceptive signals or rely on vision alone.

Each sensing modality provides distinct information about physical interaction and differs in its mounting location, output signals, and ability to capture different contact properties. Although their sensing mechanisms

Table 7 Sensing modalities used by the surveyed 53 methods.

	Wrist F/T	Joint Torque	Static Tactile
Mounting	wrist *(extrinsic/intrinsic sensing)*	each joint *(intrinsic sensing)*	gripper, fingertip, palm, or skin *(extrinsic/cutaneous)*
Raw signal	6-axis F/T vector	joint-torque vector	image, taxel array, or contact points
Derived info	end-effector wrench	Jacobian-mapped wrench; external joint torque	contact patch, pressure, shear
Limitation	no localization on the tool	needs an accurate dynamic model	limited to the sensing surface
Typical use	insertion, assembly, compliance	joint compliance	in-hand manipulation, grasp and texture sensing
Adoption	18/53 (34%)	9/53 (17%)	27/53 (51%)

have been introduced in Sections 2.1 and 2.2, here we examine the practical functionality, advantages, and limitations of each sensor type, grounded in its use across the surveyed methods. Table 7 summarizes these characteristics, while Table 6 records the sensing modalities adopted by each paper.

6.2.1 End-Effector Wrench Sensors

End-effector wrench sensors measure the resultant wrench acting at the end-effector. Typically mounted between the robot flange and the end-effector, the sensor provides a single six-axis wrench measurement. This sensing modality offers two main advantages. First, because the sensor is located near the end of the arm, it directly measures external interaction wrenches without relying on robot-dynamics models or joint-level torque estimation. Second, it can be retrofitted to a wide range of standard robotic arms, enabling wrench sensing even when the robot is not originally equipped with force/torque sensing capability. The limitations are also clear. Because the sensor provides only a single wrench measurement at the wrist, the precise contact location on the tool generally cannot be determined from the measured wrench alone. Moreover, contacts occurring elsewhere along the robot arm may remain undetected because only loads transmitted through the wrist interface are measured.

6.2.2 Joint Torque Sensors

Joint torque sensing captures interaction effects across the entire robot arm, rather than only at the end-effector. It measures the torque exerted at each actuated joint, from which two types of information can be inferred. First, the joint-torque vector can be mapped through the manipulator Jacobian to estimate the same six-axis end-effector wrench that a wrist F/T sensor provides, making end-effector contact observable even without a wrist-mounted sensor. Second, contact applied anywhere along the robot body appears in the joint-torque signal, supporting whole-arm compliance control and safe behavior under accidental link contact. The limitations are twofold. Isolating interaction torque from gravity and inertial effects requires an accurate dynamic model of the arm. In addition, joint torque sensors must be integrated into the actuators rather than added afterward, making this modality unavailable on standard position-controlled manipulators without additional hardware cost.

6.2.3 Tactile Sensors

Tactile sensors provide information about contact between the robot and its environment. Section 2.1.3 introduced the data formats used to represent tactile signals. Here, we discuss the practical specifications and selection considerations for tactile sensors. Depending on their design, these sensors can capture contact location and area, local surface geometry, pressure or force distribution, deformation, slip, and vibration. In the surveyed learning systems, the two dominant types are vision-based tactile sensors and taxel arrays. They differ in spatial resolution, sensing area, sampling rate, sensitivity, and integration requirements.

Vision-Based Tactile Sensors. A vision-based tactile sensor places one or more cameras behind a deformable contact surface. When contact occurs, the camera records changes in the shape or appearance of the surface or the movement of visual markers. The resulting tactile images can be used to estimate contact geometry, surface texture, slip, and force distribution. The main advantage of these sensors is their dense contact information and high spatial resolution, often at the submillimeter scale. However, their sensing area is generally limited to the deformable surface observed by the camera. Their sampling rate and latency also depend on the camera, image transmission, and subsequent image processing. Among the 15 surveyed methods using vision-based tactile sensing, 7 employ sensors from the GelSight family, including GelSight, GelSight Mini, and GelStereo, with one setup additionally using MC-Tac. The remaining 8 use DIGIT, STS, vision-based tactile sensors integrated into the SharpaWave hand, Xense optical tactile sensors, configurations combining multiple optical tactile sensors, or custom-built sensors.

Taxel-Based Tactile Sensors. Taxel-based sensors consist of individual sensing units arranged over a contact surface. Depending on the sensor design, each taxel records local pressure, normal or shear force, or deformation. Together, these measurements form a spatial contact map whose resolution depends on the size, spacing, and arrangement of the taxels. Compared with vision-based tactile sensors, taxel arrays can provide higher sampling rates and may be easier to distribute over curved or larger surfaces, although these properties depend on the specific sensor design.

Taxel arrays can use resistive or piezoresistive, capacitive, barometric, or magnetic sensing. In a magnetic tactile sensor, contact deforms a soft material containing a magnet or magnetic particles, which changes the magnetic field measured by nearby sensors. These changes can be used to estimate local deformation or contact forces. Magnetic tactile sensors can be compact and durable. Resistive and capacitive arrays are also widely used because they can be relatively inexpensive to fabricate and extended to larger sensing areas. Such arrays may be deployed as localized patches on gripper pads or robot links [73], or integrated into the fingertips and phalanges of dexterous robotic hands.

Vibration- and Acoustic-Based Sensing. Although not yet common among the 53 methods included in the TF-ART corpus, dynamic vibrotactile and acoustic sensing provides an additional ways of capturing contact information. Rather than measuring quasi-static deformation or pressure distributions, these sensors use accelerometers, piezoelectric elements, or high-bandwidth pressure transducers to capture short-lived vibrations and acoustic signals generated by sliding, impact, or slip. For example, BioTac encloses a conductive fluid beneath an elastomeric skin, allowing contact-induced microvibrations to propagate through the fluid as pressure waves and be measured by an internal hydro-acoustic pressure sensor [99]. The resulting temporal and spectral patterns can support slip, texture, and impact detection. While direct tactile transduction remains confined to the instrumented local surface, vibrations generated by remote contacts can propagate through the hand and a grasped object or tool, enabling learned inference of contact locations beyond that surface [107].

6.3 Challenges in Contact Sensing

Wrist-mounted wrench, joint torque, and tactile sensing capture complementary aspects of physical interaction: wrench sensing provides a direct global wrench at the end-effector, joint torque sensing extends contact visibility along the robot body, and tactile sensing focuses on the spatial location and distribution of contact at the end-effector. Together, these modalities support different levels of force, compliance, and contact-geometry awareness.

However, these modalities are rarely combined in practice. Only 4 of the 53 methods (RDP [16], TACT [31], TaF-VLA [33], MoDE-VLA [40]) pair a force-domain sensor with a tactile sensor. Most evaluations are confined to single-stage tasks on fixed robot platforms in controlled environments, while dynamic multi-contact interaction, long-horizon manipulation, and cross-platform generalization remain under-explored. Standardized evaluation protocols for tactile- and force-aware foundation models are also largely absent, although taxonomy-grounded dexterity benchmarks for anthropomorphic hands such as POMDAR [245], which score contact-rich manipulation as task throughput in both real-world and simulated settings, offer a reproducible, performance-based template that such protocols could build on. Closing these gaps is a

prerequisite for developing foundation models that acquire transferable physical interaction intelligence rather than task-specific motion patterns.

7 Concluding Discussion and Future Directions

Although recent tactile- and force-aware robot learning has achieved substantial progress, the field remains far from developing broadly transferable physical interaction intelligence. Existing studies have shown that force and tactile feedback can improve contact-rich manipulation, stabilize task execution, and complement vision-centric policies. However, the reviewed literature also exposes unresolved challenges in sensing, representation learning, policy architecture, control integration, and evaluation. We therefore divide the discussion into two parts. First, we organize the discussion around the individual components of the TF-ART pipeline and identify future research directions for each component. We then discuss broader research directions that extend beyond individual pipeline components.

7.1 Concluding Discussion on TF-ART-Centered Challenges

7.1.1 Force and Tactile Modalities

Force and tactile sensing provide complementary information about physical interaction. Force/torque measurements capture the overall interaction loads between the robot and its environment, whereas tactile sensing, as currently used in robot-learning systems, primarily provides local information about contact geometry, pressure distribution, and deformation at the contact surface. Together with vision, language, and proprioception, these signals constitute key components of the TF-ART observation space.

Despite their complementary roles, systematically integrating force and tactile sensing across different robot-learning frameworks remains challenging because of substantial cross-sensor and cross-embodiment variation. Wrist wrench and joint torque measurements are embodiment-dependent: the same physical contact can produce different readings across robot platforms because of sensor bias, sensor mounting frames, tool loads, and differences in robot dynamics and geometry. Tactile signals are similarly heterogeneous across hardware. Tactile sensors differ in sensing mechanism, spatial resolution, morphology, contact response, and mounting location, while their outputs take different forms, including tactile images, taxel maps, and contact-point measurements. Consequently, policies and encoders are often tied to particular sensors and robot embodiments, making cross-sensor and cross-embodiment generalization difficult. The field therefore lacks shared pretrained backbones comparable to those available for vision and language, leaving the complementary information provided by force and tactile sensing underutilized. The surveyed methods [16, 31, 33, 40] demonstrate the potential of combining these two modalities to obtain a more complete representation of physical interaction, but their systematic integration remains underexplored.

Future work should develop systematic methods for jointly encoding and fusing force and tactile observations across sensors and embodiments. It should also verify whether combining their complementary load-level and contact-level information consistently improves performance across diverse contact-rich tasks. Generalist tactile policies pretrained across different tactile sensors [246, 247] indicate that a shared foundation is attainable, while human-to-robot tactile alignment further demonstrates the potential for cross-embodiment policy transfer [248]. Extending such a foundation to incorporate force-domain sensing remains an open problem.

7.1.2 Representation Learning and Fusion

Representation learning and fusion together determine how multimodal observations become usable for action generation. The former aligns semantically related observations in a shared latent space while preserving fine-grained, modality-specific information, whereas the latter regulates how and when the encoded features contribute to the policy. Across the surveyed methods, these techniques help bridge cross-modal gaps and enable policies to incorporate contact information alongside vision and language [5, 23, 42, 47].

Representation learning and fusion nevertheless remain challenging in practice because different modalities vary substantially in dimensionality, temporal frequency, noise characteristics, and task-dependent relevance.

Vision and language provide effective semantic context in contact-free settings, whereas tactile and force signals become most informative during physical contact. Several surveyed methods address this phase-dependent variation in modality relevance by regulating modality contributions through expert routing [24, 40, 41] or gated filtering [11, 34, 42, 43, 45]. However, the signals that implicitly encode task phases, such as contact probabilities and learned routing weights, remain system-specific and are typically learned from task-specific demonstrations.

A promising direction is therefore to develop a generalizable phase-encoding framework. Such a framework could provide auxiliary signals throughout task execution to indicate interaction phases, such as non-contact, pre-contact, and in-contact, as well as broader task stages in general long-horizon manipulation. The resulting phase representation could guide the policy in determining which modalities are most informative, which motion primitives are most suitable, and which expert skill should be activated from an available skill library.

7.1.3 Hierarchical Policy Design

There is a fundamental tension between large-scale semantic reasoning and low-latency contact response. Robot foundation models show strong cross-task generalization, but contact-rich manipulation often requires millisecond-level feedback, millimeter-level precision, and accurate force regulation. Large policies alone may be too slow to meet the latency requirement and too coarse to achieve the required motion and force-control precision. Some surveyed systems therefore employ multi-phase architectures to address these competing requirements.

Multi-phase architectures provide a natural way to reconcile these requirements. A high-level model generates semantic or coarse motion plans, while a lower-level refinement policy or controller uses force and tactile feedback to make rapid adjustments [1, 11, 16, 31, 34, 44, 45]. Future systems should further formalize this division of labor, with slow reasoning modules handling task planning and fast reactive modules handling contact adaptation. Another promising direction is to develop dedicated low-level skill experts, such as modular in-hand manipulation skills or elementary manipulation primitives, and integrate them with high-level reasoning through skill routing. These skill-level experts should not be tied to specific object-task pairs. Instead, they should generalize across objects while specializing in particular motion patterns, such as approaching, pushing, or avoiding obstacles.

7.1.4 Integrating Learned Policies with Robot Control

Learning-based methods can model complex action distributions and learn from demonstrations, but contact-rich execution also depends on model-based controllers that provide stability and safety. Current systems often connect learned policies to low-level controllers mainly through action commands, leaving other control-relevant parameters, such as reference force and stiffness, to be tuned manually through experiments. Several works have explored more flexible integration by allowing learning models to predict these control parameters or controller-specific targets, which are referred to as intermediate modalities in [1, 2, 6, 13, 18, 34, 37, 44, 249]. Since predicted parameters are meaningful only when the robot exposes a matching control mode, the trained models are inherently tied to the corresponding platforms.

To reduce this platform dependence, a promising direction is to develop action-generation policies that mimic compliant-control behavior for low-cost robots without robust force-sensing or force-control capabilities. Such policies could leverage robot-dynamics-based force prediction, operate under position-control interfaces, and generalize across a wider range of platforms and contact-rich tasks.

7.1.5 Tasks, Benchmarks, and Evaluation

Evaluation protocols are still insufficient for assessing generalizable physical interaction capability. Many existing studies focus on short-horizon, single-stage tasks such as insertion, wiping, and pick-and-place, while dynamic multi-contact manipulation, deformable-object handling, and long-horizon household tasks are underexplored. Moreover, experiments are often conducted on fixed platforms with specific sensor configurations, making it difficult to compare methods systematically beyond task success rate or to evaluate cross-embodiment transfer.

The field therefore needs standardized tactile- and force-aware benchmarks that cover diverse objects, robot platforms, and sensor configurations. Recent taxonomy-grounded dexterity benchmarks discussed in Section 6 illustrate one route toward such reproducible, performance-based evaluation. Future benchmarks should measure not only task success, but also force safety, contact stability, recovery from perturbations, generalization to unseen materials, and robustness under sensor failure. Nevertheless, assembling data collected from multiple platforms and sensors into a unified benchmark remains highly challenging, requiring substantial effort from both methodological and engineering perspectives.

7.2 Broader Future Directions

Beyond the component-wise directions summarized under the proposed TF-ART, we further discuss broader directions that are not yet fully developed in the tactile- and force-aware manipulation context, but are necessary and promising for future research.

Simulation and Data Scaling. Simulation and data scaling remain major bottlenecks. Large-scale robot foundation models benefit from broad and diverse datasets, but tactile and force data are expensive to collect and difficult to simulate accurately. Current simulators still struggle to reproduce high-resolution tactile deformation, frictional contact, material compliance, and multi-contact dynamics. Future research should combine real-world data collection, physics-based simulation, and tactile and force prediction to reduce data-collection costs and enable data scaling in simulation environments [45, 49, 176]. In particular, real-world data can be used to calibrate and guide more accurate simulation of visual-tactile images, taxel-array deformation, external wrench measurements, and internal force or torque perception. This would support the generation of large-scale synthetic data that better approximates real-world contact phenomena, thereby reducing the sim-to-real gap and enabling more direct real-world deployment of tactile- and force-aware models.

Contact-informed World Action Models. Another promising direction is to connect tactile- and force-aware robot learning with recent advances in World Action Models (WAMs) [176, 185, 250, 251, 252, 253, 254, 255]. Most existing WAMs use video-based latent prediction as their main objective, but they rarely formulate future-state prediction from an interaction-centered perspective. In contact-rich manipulation, a world model should predict not only the visual evolution of the environment, but also object deformation, dynamic behavior, and the spatial relationship between the robot and the environment under contact. In this way, WAMs could move toward more general-purpose world simulators that equip robots with physically grounded reasoning capabilities. Such models may allow robots to anticipate interaction outcomes and respond to predicted, or "imagined," forces even when direct force sensing is unavailable. A more practical near-term direction is to jointly predict visual, tactile, and force evolution instead of modeling the full environment dynamics. This could help robots anticipate contact events, avoid force overshoot, and establish contact more reliably.

Touch and Force as Intrinsic Critics. Building on these predictive capabilities, a further promising direction is to use tactile and force sensing as a primary substrate for robot self-supervision and self-improvement, rather than merely as auxiliary feedback. This direction reflects a broader shift from merely generating robot behaviors to continuously evaluating, correcting, and improving them, as exemplified by recent work on test-time policy steering [256], verifier-guided action selection [257, 258], self-correcting and contact-aware world models [252, 259, 260], force-based anomaly detection [261], and robotic reward modeling [262, 263]. In contact-rich manipulation, tactile and force signals are particularly well suited to this role because they reveal whether an interaction is physically correct, rather than merely visually plausible. Future systems could verify candidate actions against predicted contact trajectories, use discrepancies between expected and observed signals to detect failures and refine policies or world models online, and train contact-aware reward models that assess execution quality, stability, and recovery from both successful and failed experiences. In this view, touch and force are not only perception channels or control inputs, but intrinsic critics by which robots can assess and improve the physical quality of their own execution.

Mechanical Compliance as a Design Variable. A further underexplored direction is to treat mechanical compliance as a design variable rather than only a control objective. The control methods surveyed in Section 5 regulate compliance through impedance, admittance, and force control on largely rigid manipulators. Given the tight coupling between learned policies and platform control interfaces discussed above, an orthogonal route is to embed compliance directly in the hardware, through soft or tendon-driven actuation, antagonistic variable-stiffness joints, and inherently compliant anthropomorphic hands [264, 265, 266]. Such embodiments

absorb impact, distribute contact, and stabilize interaction passively, reducing the bandwidth and accuracy demanded of force sensing and feedback control, and in some cases enabling safe contact-rich behavior with little or no explicit force sensing. This points toward joint co-design of body and policy, in which morphology, actuation compliance, and learned control are optimized together rather than layering a policy onto a fixed platform. For tactile- and force-aware learning specifically, mechanically compliant embodiments also reshape the sensing problem, since intrinsic compliance changes how contact forces are transmitted to and observed by the available sensors.

In the long term, tactile- and force-grounded robot intelligence should move beyond learning task-specific motion patterns and toward acquiring transferable physical concepts, such as contact, pressure, friction, and stability. Achieving this goal will require unified multimodal representations, phase-aware policy architectures, systematic policy-control integration, compliant embodiment and body-policy co-design, stronger awareness of world evolution, and standardized benchmarks for real-world physical interaction.

References

[1] Peiyuan Zhi, Peiyang Li, Jianqin Yin, Baoxiong Jia, and Siyuan Huang. Learning a unified policy for position and force control in legged loco-manipulation, 2025. URL https://arxiv.org/abs/2505.20829.

[2] Tifanny Portela, Gabriel B Margolis, Yandong Ji, and Pulkit Agrawal. Learning force control for legged manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15366–15372. IEEE, 2024.

[3] Michael Noseworthy, Bingjie Tang, Bowen Wen, Ankur Handa, Chad Kessens, Nicholas Roy, Dieter Fox, Fabio Ramos, Yashraj Narang, and Iretiayo Akinola. Forge: Force-guided exploration for robust contact-rich manipulation under uncertainty. *IEEE Robotics and Automation Letters*, 2025.

[4] Zongzheng Zhang, Haobo Xu, Zhuo Yang, Chenghao Yue, Zehao Lin, Huan-ang Gao, Ziwei Wang, and Hao Zhao. Ta-vla: Elucidating the design space of torque-aware vision-language-action models. *arXiv preprint arXiv:2509.07962*, 2025.

[5] Carmelo Sferrazza, Younggyo Seo, Hao Liu, Youngwoon Lee, and Pieter Abbeel. The power of the senses: Generalizable manipulation from vision and touch through masked multimodal learning. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9698–9705. IEEE, 2024.

[6] Yifan Hou, Zeyi Liu, Cheng Chi, Eric Cousineau, Naveen Kuppuswamy, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Adaptive compliance policy: Learning approximate compliance for diffusion guided control. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4829–4836. IEEE, 2025.

[7] Bo Zhou, Ruixuan Jiao, Yi Li, Xiaogang Yuan, Fang Fang, and Shihua Li. Admittance visuomotor policy learning for general-purpose contact-rich manipulations. *IEEE Transactions on Industrial Electronics*, 2025.

[8] Jianlan Luo, Charles Xu, Jeffrey Wu, and Sergey Levine. Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning. *Science Robotics*, 10(105):eads5033, 2025.

[9] Thanpimon Buamanee, Masato Kobayashi, Yuki Uranishi, and Haruo Takemura. Bi-act: Bilateral control-based imitation learning via action chunking with transformer. In *2024 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 410–415. IEEE, 2024.

[10] Claire Chen, Zhongchun Yu, Hojung Choi, Mark Cutkosky, and Jeannette Bohg. Dexforce: Extracting force-informed actions from kinesthetic demonstrations for dexterous manipulation. *IEEE Robotics and Automation Letters*, 2025.

[11] Zihao He, Hongjie Fang, Jingjing Chen, Hao-Shu Fang, and Cewu Lu. Foar: Force-aware reactive policy for contact-rich robotic manipulation. *IEEE Robotics and Automation Letters*, 2025.

[12] Wenhai Liu, Junbo Wang, Yiming Wang, Weiming Wang, and Cewu Lu. Forcemimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1105–1112. IEEE, 2025.

[13] Malek Aburub, Cristian C Beltran-Hernandez, Tatsuya Kamijo, and Masashi Hamaya. Learning diffusion policies from demonstrations for compliant contact-rich manipulation. In *2026 IEEE/SICE International Symposium on System Integration (SII)*, pages 28–34. IEEE, 2026.

[14] Tatsuya Kamijo, Cristian C Beltran-Hernandez, and Masashi Hamaya. Learning variable compliance control from a few demonstrations for bimanual robot with haptic feedback teleoperation system. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 12663–12670. IEEE, 2024.

[15] Toru Lin, Yu Zhang, Qiyang Li, Haozhi Qi, Brent Yi, Sergey Levine, and Jitendra Malik. Learning visuotactile skills with two multifingered hands. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5637–5643. IEEE, 2025.

[16] Han Xue, Jieji Ren, Wendi Chen, Gu Zhang, Yuan Fang, Guoying Gu, Huazhe Xu, and Cewu Lu. Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation. *arXiv preprint arXiv:2503.02881*, 2025.

[17] Jeon Ho Kang, Sagar Joshi, Ruopeng Huang, and Satyandra K Gupta. Robotic compliant object prying using diffusion policy guided by vision and force observations. *IEEE Robotics and Automation Letters*, 2025.

[18] Yansong Wu, Zongxie Chen, Fan Wu, Lingyun Chen, Liding Zhang, Zhenshan Bing, Abdalla Swikir, Sami Haddadin, and Alois Knoll. Tacdiffusion: Force-domain diffusion policy for precise tactile manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11831–11837. IEEE, 2025.

[19] Erik Helmut, Niklas Funk, Tim Schneider, Cristiana de Farias, and Jan Peters. Tactile-conditioned diffusion policy for force-aware robotic manipulation. *arXiv preprint arXiv:2510.13324*, 2025.

[20] Liang Heng, Haoran Geng, Kaifeng Zhang, Pieter Abbeel, and Jitendra Malik. Vitacformer: Learning cross-modal representation for visuo-tactile dexterous manipulation. *arXiv preprint arXiv:2506.15953*, 2025.

[21] Binghao Huang, Yixuan Wang, Xinyi Yang, Yiyue Luo, and Yunzhu Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. *arXiv preprint arXiv:2410.24091*, 2024.

[22] Zhengtong Xu, Raghava Uppuluri, Xinwei Zhang, Cael Fitch, Philip Glen Crandall, Wan Shou, Dongyi Wang, and Yu She. Unit: Data efficient tactile representation with generalization to unseen objects. *IEEE Robotics and Automation Letters*, 2025.

[23] Joshua Jones, Oier Mees, Carmelo Sferrazza, Kyle Stachowicz, Pieter Abbeel, and Sergey Levine. Beyond sight: Finetuning generalist robot policies with heterogeneous sensors via language grounding. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5961–5968. IEEE, 2025.

[24] Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, et al. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. *arXiv preprint arXiv:2505.22159*, 2025.

[25] Jialei Huang, Shuo Wang, Fanqi Lin, Yihang Hu, Chuan Wen, and Yang Gao. Tactile-vla: unlocking vision-language-action model's physical knowledge for tactile generalization. *arXiv preprint arXiv:2507.09160*, 2025.

[26] Peng Hao, Chaofan Zhang, Dingzhe Li, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. Tla: Tactile-language-action model for contact-rich manipulation. *arXiv preprint arXiv:2503.08548*, 2025.

[27] Haizhou Ge, Yufei Jia, Zheng Li, Yue Li, Zhixing Chen, Ruqi Huang, and Guyue Zhou. Filic: Dual-loop force-guided imitation learning with impedance torque control for contact-rich manipulation tasks. *arXiv preprint arXiv:2509.17053*, 2025.

[28] Geonhyup Lee, Yeongjin Lee, Kangmin Kim, Seongju Lee, Sangjun Noh, Seunghyeok Back, and Kyoobin Lee. Manipforce: Force-guided policy learning with frequency-aware representation for contact-rich manipulation. *arXiv preprint arXiv:2509.19047*, 2025.

[29] Yuki Shirai, Kei Ota, Devesh K Jha, and Diego Romeres. Sim-to-real contact-rich pivoting via optimization-guided rl with vision and touch. In *NeurIPS 2025 Workshop on Embodied World Models for Decision Making*, 2025.

[30] Zifan Zhao, Siddhant Haldar, Jinda Cui, Lerrel Pinto, and Raunaq Bhirangi. Touch begins where vision ends: Generalizable policies for contact-rich manipulation. *arXiv preprint arXiv:2506.13762*, 2025.

[31] Masaki Murooka, Takahiro Hoshi, Kensuke Fukumitsu, Shimpei Masuda, Marwan Hamze, Tomoya Sasaki, Mitsuharu Morisawa, and Eiichi Yoshida. Tact: Humanoid whole-body contact manipulation through deep imitation learning with tactile modality. *IEEE Robotics and Automation Letters*, 2025.

[32] Qi Ye, Qingtao Liu, Siyun Wang, Jiaying Chen, Yu Cui, Ke Jin, Huajin Chen, Xuan Cai, Gaofeng Li, and Jiming Chen. Visual-tactile pretraining and online multitask learning for humanlike manipulation dexterity. *Science Robotics*, 11(110):eady2869, 2026.

[33] Yuzhe Huang, Pei Lin, Wanlin Li, Daohan Li, Jiajun Li, Jiaming Jiang, Chenxi Xiao, and Ziyuan Jiao. Tactile-force alignment in vision-language-action models for force-aware manipulation. *arXiv preprint arXiv:2601.20321*, 2026.

[34] Hongjie Fang, Shirun Tang, Mingyu Mei, Haoxiang Qin, Zihao He, Jingjing Chen, Ying Feng, Chenxi Wang, Wanxi Liu, Zaixing He, et al. Force policy: Learning hybrid force-position control policy under interaction frame for contact-rich manipulation. *arXiv preprint arXiv:2602.22088*, 2026.

[35] Jason Jingzhou Liu, Yulong Li, Kenneth Shaw, Tony Tao, Ruslan Salakhutdinov, and Deepak Pathak. Factr: Force-attending curriculum training for contact-rich policy learning. *arXiv preprint arXiv:2502.17432*, 2025.

[36] Trevor Ablett, Oliver Limoyo, Adam Sigal, Affan Jilani, Jonathan Kelly, Kaleem Siddiqi, Francois Hogan, and Gregory Dudek. Multimodal and force-matched imitation learning with a see-through visuotactile sensor. *IEEE Transactions on Robotics*, 41:946–959, 2024.

[37] Jeremy A Collins, Cody Houff, You Liang Tan, and Charles C Kemp. Forcesight: Text-guided mobile manipulation with visual-force goals. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10874–10880. IEEE, 2024.

[38] Yike Zhang, Yaonan Wang, Xinxin Sun, Kaizhen Huang, Zhiyuan Xu, Junjie Ji, Zhengping Che, Jian Tang, and Jingtao Sun. Craft: Adapting vla models to contact-rich manipulation via force-aware curriculum fine-tuning. *arXiv preprint arXiv:2602.12532*, 2026.

[39] Joohwan Seo, Arvind Kruthiventy, Soomi Lee, Megan Teng, Seoyeon Choi, Xiang Zhang, Jongeun Choi, and Roberto Horowitz. Equicontact: A hierarchical se (3) vision-to-force equivariant policy for spatially generalizable contact-rich tasks. *arXiv preprint arXiv:2507.10961*, 2025.

[40] Tutian Tang, Xingyu Ji, Wanli Xing, Ce Hao, Wenqiang Xu, Lin Shao, Cewu Lu, Qiaojun Yu, Jiangmiao Pang, and Kaifeng Zhang. Towards human-like manipulation through rl-augmented teleoperation and mixture-of-dexterous-experts vla. *arXiv preprint arXiv:2603.08122*, 2026.

[41] Yang Li, Hongru Jiang, Junjie Xia, Hongquan Zhang, Jinda Du, Yunsong Zhou, Jia Zeng, Ce Hao, Jieji Ren, Qiaojun Yu, et al. Forcevla2: Unleashing hybrid force-position control with force awareness for contact-rich manipulation. *arXiv preprint arXiv:2603.15169*, 2026.

[42] Minchi Ruan, LiangQing Zhou, Hongtong Li, Zongtao Wang, ZhaoMing Lu, Jianwei Zhang, and Bin Fang. Retac-act: A state-gated vision-tactile fusion transformer for precision assembly. *arXiv preprint arXiv:2603.09565*, 2026.

[43] Kaidi Zhang, Heng Zhang, Zhengtong Xu, Zhiyuan Zhang, Md Rakibul Islam Prince, Xiang Li, Xiaojing Han, Yuhao Zhou, Arash Ajoudani, and Yu She. Tacvla: Contact-aware tactile fusion for robust vision-language-action manipulation. *arXiv preprint arXiv:2603.12665*, 2026.

[44] Di Zhang, Chengbo Yuan, Chuan Wen, Hai Zhang, Junqiao Zhao, and Yang Gao. Kinedex: Learning tactile-informed visuomotor policies via kinesthetic teaching for dexterous manipulation. *arXiv preprint arXiv:2505.01974*, 2025.

[45] Yuhang Zheng, Songen Gu, Weize Li, Yupeng Zheng, Yujie Zang, Shuai Tian, Xiang Li, Ce Hao, Chen Gao, Si Liu, et al. Omnivta: Visuo-tactile world modeling for contact-rich robotic manipulation. *arXiv preprint arXiv:2603.19201*, 2026.

[46] Zhengxue Cheng, Yiqian Zhang, Wenkang Zhang, Haoyu Li, Keyu Wang, Li Song, and Hengdi Zhang. Omnivtla: Vision-tactile-language-action model with semantic-aligned tactile sensing. *arXiv preprint arXiv:2508.08706*, 2025.

[47] Yufeng Tian, Shuiqi Cheng, Tianming Wei, Tianxing Zhou, Yuanhang Zhang, Zixian Liu, Qianwei Han, Zhecheng Yuan, and Huazhe Xu. Vitas: Visual tactile soft fusion contrastive learning for visuomotor learning. *arXiv preprint arXiv:2602.11643*, 2026.

[48] Jianxin Bi, Kevin Yuchen Ma, Ce Hao, Mike Zheng Shou, and Harold Soh. Vla-touch: Enhancing vision-language-action models with dual-level tactile feedback. *arXiv preprint arXiv:2507.17294*, 2025.

[49] Tianhao Wu, Jinzhou Li, Jiyao Zhang, Mingdong Wu, and Hao Dong. Canonical representation and force-based pretraining of 3d tactile for dexterous visuo-tactile policy learning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6786–6792. IEEE, 2025.

[50] Zhengnan Sun, Zhaotai Shi, Jiayin Chen, Qingtao Liu, Yu Cui, Jiming Chen, and Qi Ye. Vtao-bimanip: Masked visual-tactile-action pre-training with object understanding for bimanual dexterous manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3201–3208. IEEE, 2025.

[51] Chaofan Zhang, Peng Hao, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. Vtla: Vision-tactile-language-action model with preference learning for insertion manipulation, 2025. URL https://arxiv.org/abs/2505.09577.

[52] Jiankai Sun, Aidan Curtis, Yang You, Yan Xu, Michael Koehle, Qianzhong Chen, Suning Huang, Leonidas Guibas, Sachin Chitta, Mac Schwager, et al. Arch: Hierarchical hybrid learning for long-horizon contact-rich robotic assembly. *arXiv preprint arXiv:2409.16451*, 2024.

[53] Zhemeng Zhang, Jiahua Ma, Xincheng Yang, Xin Wen, Yuzhi Zhang, Boyan Li, Yiran Qin, Jin Liu, Can Zhao, Li Kang, et al. Touchguide: Inference-time steering of visuomotor policies via touch guidance. *arXiv preprint arXiv:2601.20239*, 2026.

[54] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[55] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[56] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[57] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi-0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[58] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. pi-0.5: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[59] Yuxin Ray Song, Jinzhou Li, Rao Fu, Devin Murphy, Kaichen Zhou, Rishi Shiv, Yaqi Li, Haoyu Xiong, Crystal Elaine Owens, Yilun Du, et al. Opentouch: Bringing full-hand touch to real-world interaction. *arXiv preprint arXiv:2512.16842*, 2025.

[60] Marc H Raibert and John J Craig. Hybrid position/force control of manipulators. 1981.

[61] Neville Hogan. Impedance control: An approach to manipulation: Part ii—implementation. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):8–16, March 1985. ISSN 0022-0434. doi: 10.1115/1.3140713.

[62] O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, February 1987. ISSN 2374-8710. doi: 10.1109/JRA.1987.1087068.

[63] Arvid QL Keemink, Herman Van der Kooij, and Arno HA Stienen. Admittance control for physical human–robot interaction. *The International Journal of Robotics Research*, 37(11):1421–1444, 2018.

[64] Howard R. Nicholls and Mark H. Lee. A survey of robot tactile sensing technology. *The International Journal of Robotics Research*, 8(3):3–30, 1989. doi: 10.1177/027836498900800301.

[65] Mark H. Lee and Howard R. Nicholls. Tactile sensing for mechatronics—a state of the art survey. *Mechatronics*, 9(1):1–31, 1999. doi: 10.1016/S0957-4158(98)00045-2.

[66] Johan Tegin and Jan Wikander. Tactile sensing in intelligent robotic manipulation—a review. *Industrial Robot: An International Journal*, 32(1):64–70, 2005. doi: 10.1108/01439910510573318.

[67] Ravinder S. Dahiya, Giorgio Metta, Maurizio Valle, and Giulio Sandini. Tactile sensing—from humans to humanoids. *IEEE Transactions on Robotics*, 26(1):1–20, 2010. doi: 10.1109/TRO.2009.2033627.

[68] Hanna Yousef, Mehdi Boukallel, and Kaspar Althoefer. Tactile sensing for dexterous in-hand manipulation in robotics—a review. *Sensors and Actuators A: Physical*, 167(2):171–187, 2011. doi: 10.1016/j.sna.2011.02.038.

[69] Pedro Silva Girão, Pedro Miguel Pinto Ramos, Octavian Postolache, and José Miguel Dias Pereira. Tactile sensors for robotic applications. *Measurement*, 46(3):1257–1271, 2013. doi: 10.1016/j.measurement.2012.11.015.

[70] Ravinder S. Dahiya, Philipp Mittendorfer, Maurizio Valle, Gordon Cheng, and Vladimir J. Lumelsky. Directions toward effective utilization of tactile skin: A review. *IEEE Sensors Journal*, 13(11):4121–4138, 2013. doi: 10.1109/JSEN.2013.2279056.

[71] Zhanat Kappassov, Juan-Antonio Corrales, and Véronique Perdereau. Tactile sensing in dexterous robot hands—review. *Robotics and Autonomous Systems*, 74:195–220, 2015. doi: 10.1016/j.robot.2015.07.015.

[72] Peter Roberts, Mason Zadan, and Carmel Majidi. Soft tactile sensing skins for robotics. *Current Robotics Reports*, 2:343–354, 2021. doi: 10.1007/s43154-021-00065-2.

[73] Gordon Cheng, Emmanuel Dean-Leon, Florian Bergner, Julio Rogelio Guadarrama Olvera, Quentin Leboutet, and Philipp Mittendorfer. A comprehensive realization of robot skin: Sensors, sensing, control, and applications. *Proceedings of the IEEE*, 107(10):2034–2051, 2019.

[74] Shan Luo, Joao Bimbo, Ravinder Dahiya, and Hongbin Liu. Robotic tactile perception of object properties: A review. *Mechatronics*, 48:54–67, 2017. doi: 10.1016/j.mechatronics.2017.11.002.

[75] Qiang Li, Oliver Kroemer, Zhe Su, Filipe Fernandes Veiga, Mohsen Kaboli, and Helge Joachim Ritter. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics*, 36(6):1619–1634, 2020. doi: 10.1109/TRO.2020.3003230.

[76] Nicolás Navarro-Guerrero, Sibel Toprak, Josip Josifovski, and Lorenzo Jamone. Visuo-haptic object perception for robots: An overview. *Autonomous Robots*, 47(4):377–403, 2023. doi: 10.1007/s10514-023-10091-y.

[77] Shoujie Li, Zihan Wang, Changsheng Wu, Xiang Li, Shan Luo, Bin Fang, Fuchun Sun, Xiao-Ping Zhang, and Wenbo Ding. When vision meets touch: A contemporary review for visuotactile sensors from the signal processing perspective. *IEEE Journal of Selected Topics in Signal Processing*, 18(3):267–287, 2024. doi: 10.1109/JSTSP.2024.3416841.

[78] Alessandro Albini, Mohsen Kaboli, Giorgio Cannata, and Perla Maiolino. Representing data in robotic tactile perception—a review. *arXiv preprint arXiv:2510.10804*, 2025. doi: 10.48550/arXiv.2510.10804. URL https://arxiv.org/abs/2510.10804.

[79] Lucia Seminara, Paolo Gastaldo, Simon J. Watt, Kenneth F. Valyear, Fernando Zuher, and Fulvio Mastrogiovanni. Active haptic perception in robots: A review. *Frontiers in Neurorobotics*, 13:53, 2019. doi: 10.3389/fnbot.2019.00053.

[80] Enrico Donato, Matteo Lo Preti, Lucia Beccai, and Egidio Falotico. Sensorimotor control strategies for tactile robotics. *arXiv preprint arXiv:2501.09468*, 2025. doi: 10.48550/arXiv.2501.09468. URL https://arxiv.org/abs/2501.09468.

[81] Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. *Journal of machine learning research*, 22(30):1–82, 2021.

[82] Harish Ravichandar, Athanasios S Polydoros, Sonia Chernova, and Aude Billard. Recent advances in robot learning from demonstration. *Annual review of control, robotics, and autonomous systems*, 3(1):297–330, 2020.

[83] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J Andrew Bagnell, Pieter Abbeel, and Jan Peters. An algorithmic perspective on imitation learning. *Foundations and Trends® in Robotics*, 7(1-2):1–179, 2018.

[84] Maryam Zare, Parham M Kebria, Abbas Khosravi, and Saeid Nahavandi. A survey of imitation learning: Algorithms, recent developments, and challenges. *IEEE Transactions on Cybernetics*, 54(12):7173–7186, 2024.

[85] Julen Urain, Ajay Mandlekar, Yilun Du, Nur Muhammad, Danfei Xu, Katerina Fragkiadaki, Georgia Chalvatzaki, Jan Peters, et al. A survey on deep generative models for robot learning from multimodal demonstrations. *IEEE Transactions on Robotics*, 42:60–79, 2025.

[86] Markku Suomalainen, Yiannis Karayiannidis, and Ville Kyrki. A survey of robot manipulation in contact. *Robotics and Autonomous Systems*, 156:104224, 2022.

[87] Toshiaki Tsuji, Yasuhiro Kato, Gokhan Solak, Heng Zhang, Tadej Petrič, Francesco Nori, and Arash Ajoudani. A survey on imitation learning for contact-rich tasks in robotics. *The International Journal of Robotics Research*, page 02783649261417694, 2025.

[88] Gaofeng Li, Ruize Wang, Peisen Xu, Qi Ye, and Jiming Chen. The developments and challenges toward dexterous and embodied robotic manipulation: A survey. *IEEE Robotics & Automation Magazine*, 33(1):24–38, 2026. doi: 10.1109/MRA.2025.3642671.

[89] Shan An, Ziyu Meng, Chao Tang, Yuning Zhou, Tengyu Liu, Fangqiang Ding, Shufang Zhang, Yao Mu, Ran Song, Wei Zhang, et al. Dexterous manipulation through imitation learning: A survey. *arXiv preprint arXiv:2504.03515*, 2025.

[90] Edgar Welte and Rania Rayyes. Interactive imitation learning for dexterous robotic manipulation: challenges and perspectives—a survey. *Frontiers in Robotics and AI*, 12:1682437, 2025.

[91] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 44(5):701–739, 2025.

[92] Yafei Hu, Quanting Xie, Vidhi Jain, Jonathan Francis, Jay Patrikar, Nikhil Keetha, Seungchan Kim, Yaqi Xie, Tianyi Zhang, Hao-Shu Fang, et al. Toward general-purpose robots via foundation models: A survey and meta-analysis. *arXiv preprint arXiv:2312.08782*, 2023.

[93] William Xie and Nikolaus Correll. Towards forceful robotic foundation models: a literature survey. *arXiv preprint arXiv:2504.11827*, 2025.

[94] Shan Luo, Nathan F Lepora, Wenzhen Yuan, Kaspar Althoefer, Gordon Cheng, and Ravinder Dahiya. Tactile robotics: An outlook. *IEEE Transactions on Robotics*, 2025.

[95] James J Gibson. Observations on active touch. *Psychological review*, 69(6):477, 1962.

[96] Tony J Prescott, Mathew E Diamond, and Alan M Wing. Active touch sensing. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 366(1581):2989, 2011.

[97] Peter C Gaston and Tomas Lozano-Perez. Tactile recognition and localization using object models: The case of polyhedra on a plane. *IEEE transactions on pattern analysis and machine intelligence*, (3):257–266, 1984.

[98] Allison M Okamura and Mark R Cutkosky. Feature detection for haptic exploration with robotic fingers. *The International Journal of Robotics Research*, 20(12):925–938, 2001.

[99] Jeremy A Fishel and Gerald E Loeb. Bayesian exploration for intelligent identification of textures. *Frontiers in neurorobotics*, 6:4, 2012.

[100] Nathan F Lepora, Uriel Martinez-Hernandez, and Tony J Prescott. Active bayesian perception for simultaneous object localization and identification. In *Robotics: Science and Systems*, pages 1–8, 2013.

[101] Robert D Howe, Mark R Cutkosky, et al. Sensing skin acceleration for slip and texture perception. In *ICRA*, pages 145–150, 1989.

[102] Robert D Howe and Mark R Cutkosky. Dynamic tactile sensing: Perception of fine surface features with stress rate sensing. *IEEE transactions on robotics and automation*, 9(2):140–151, 1993.

[103] Tasbolat Taunyazov, Weicong Sng, Hian Hian See, Brian Lim, Jethro Kuan, Abdul Fatir Ansari, Benjamin CK Tee, and Harold Soh. Event-driven visual-tactile sensing and learning for robots. *arXiv preprint arXiv:2009.07083*, 2020.

[104] Vincent Hayward. Is there a "Plenhaptic" function? *Philosophical Transactions of the Royal Society B: Biological Sciences*, 366(1581):3115–3122, 2011. doi: 10.1098/rstb.2011.0150.

[105] Yitian Shao, Vincent Hayward, and Yon Visell. Spatial patterns of cutaneous vibration during whole-hand haptic interactions. *Proceedings of the National Academy of Sciences*, 113(15):4188–4193, 2016.

[106] Luke E Miller, Luca Montroni, Eric Koun, Romeo Salemme, Vincent Hayward, and Alessandro Farnè. Sensing with tools extends somatosensory processing beyond the body. *Nature*, 561(7722):239–242, 2018.

[107] Tasbolat Taunyazov, Luar Shui Song, Eugene Lim, Hian Hian See, David Lee, Benjamin CK Tee, and Harold Soh. Extended tactile perception: Vibration sensing through tools and grasped objects. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1755–1762. IEEE, 2021.

[108] Harold Soh, Yanyu Su, and Yiannis Demiris. Online spatio-temporal gaussian process experts with application to tactile classification. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4489–4496. IEEE, 2012.

[109] Harold Soh and Yiannis Demiris. Incrementally learning objects by touch: Online discriminative and generative models for tactile-based recognition. *IEEE transactions on haptics*, 7(4):512–525, 2014.

[110] Mohsen Kaboli and Gordon Cheng. Robust tactile descriptors for discriminating objects from textural properties via artificial robotic skin. *IEEE Transactions on Robotics*, 34(4):985–1003, 2018.

[111] Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H Adelson, and Sergey Levine. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3300–3307, 2018.

[112] Michelle A Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks. *IEEE Transactions on Robotics*, 36(3):582–596, 2020.

[113] Joseph M Romano, Kaijen Hsiao, Günter Niemeyer, Sachin Chitta, and Katherine J Kuchenbecker. Human-inspired robotic grasp control with tactile sensing. *IEEE Transactions on Robotics*, 27(6):1067–1079, 2011.

[114] Klas Kronander and Aude Billard. Learning compliant manipulation through kinesthetic and tactile human-robot interaction. *IEEE transactions on haptics*, 7(3):367–380, 2013.

[115] Nathan F Lepora, Kirsty Aquilina, and Luke Cramphorn. Exploratory tactile servoing with active touch. *IEEE Robotics and Automation Letters*, 2(2):1156–1163, 2017.

[116] Francois R Hogan, Jose Ballester, Siyuan Dong, and Alberto Rodriguez. Tactile dexterity: Manipulation primitives with tactile feedback. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 8863–8869. IEEE, 2020.

[117] Michael C Welle, Martina Lippi, Haofei Lu, Jens Lundell, Andrea Gasparri, and Danica Kragic. Enabling robot manipulation of soft and rigid objects with vision-based tactile sensors. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, pages 1–7. IEEE, 2023.

[118] Antonio Bicchi. A criterion for optimal design of multi-axis force sensors. *Robotics and Autonomous Systems*, 10 (4):269–286, 1992.

[119] Sheng A Liu and Hung L Tzo. A novel six-component force sensor of good measurement isotropy and sensitivities. *Sensors and Actuators A: Physical*, 100(2-3):223–230, 2002.

[120] Farhad Aghili, Martin Buehler, and John M Hollerbach. Design of a hollow hexaform torque sensor for robot joints. *The International Journal of Robotics Research*, 20(12):967–976, 2001.

[121] Alin Albu-Schäffer, Sami Haddadin, Ch Ott, Andreas Stemmer, Thomas Wimböck, and Gerhard Hirzinger. The dlr lightweight robot: design and control concepts for robots in human environments. *Industrial Robot: an international journal*, 34(5):376–385, 2007.

[122] Wisama Khalil and Etienne Dombre. *Modeling, identification and control of robots*. Butterworth-Heinemann, 2004.

[123] Jin Hu and Rong Xiong. Contact force estimation for robot manipulator using semiparametric model and disturbance kalman filter. *IEEE Transactions on Industrial Electronics*, 65(4):3365–3375, 2017.

[124] Shilin Shan and Quang-Cuong Pham. Fast payload calibration for sensorless contact estimation using model pre-training. *IEEE Robotics and Automation Letters*, 9(10):9007–9014, 2024. doi: 10.1109/LRA.2024.3455800.

[125] Shilin Shan and Quang-Cuong Pham. Sensorless estimation of contact using deep-learning for human-robot interaction. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12935–12941, 2024. doi: 10.1109/ICRA57147.2024.10609990.

[126] Nural Yilmaz, Jie Ying Wu, Peter Kazanzides, and Ugur Tumerdem. Neural network based inverse dynamics identification and external force estimation on the da vinci research kit. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1387–1393. IEEE, 2020.

[127] Zhiyang Dou, John U Onyemelukwe, Hangxing Zhang, Heng Zhang, Minghao Guo, Yunsheng Tian, Michal Piotr Lipiec, Joshua Jacob, Chao Liu, Peter Yichen Chen, et al. Neuralactuator: Neural actuation modeling for robot dynamics and external force perception. *arXiv preprint arXiv:2607.11734*, 2026.

[128] Wenzhen Yuan, Siyuan Dong, and Edward H Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.

[129] Changyi Lin, Han Zhang, Jikai Xu, Lei Wu, and Huazhe Xu. 9dtact: A compact vision-based tactile sensor for accurate 3d shape reconstruction and generalizable 6d force estimation. *IEEE Robotics and Automation Letters*, 9(2):923–930, 2023.

[130] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[131] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[132] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[133] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[134] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[135] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[136] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[137] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021.

[138] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17868–17879, 2024.

[139] Wenzhong Guo, Jianwen Wang, and Shiping Wang. Deep multimodal representation learning: A survey. *Ieee Access*, 7:63373–63394, 2019.

[140] Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Foundations & trends in multimodal machine learning: Principles, challenges, and open questions. *ACM computing surveys*, 56(10):1–42, 2024.

[141] Yongshuo Zong, Oisin Mac Aodha, and Timothy M Hospedales. Self-supervised multimodal learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(7):5299–5318, 2024.

[142] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[143] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[144] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[145] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[146] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.

[147] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[148] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[149] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014.

[150] Mike Wu and Noah Goodman. Multimodal generative models for scalable weakly-supervised learning. *Advances in neural information processing systems*, 31, 2018.

[151] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

[152] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[153] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[154] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

[155] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[156] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, et al. Openvla: An open-source vision-language-action model. In *8th Annual Conference on Robot Learning*.

[157] Wenxuan Song, Ziyang Zhou, Han Zhao, Jiayi Chen, Pengxiang Ding, Haodong Yan, Yuxin Huang, Feilong Tang, Donglin Wang, and Haoang Li. Reconvla: Reconstructive vision-language-action model as effective robot perceiver. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

[158] Jinming Li, Yichen Zhu, Zhibin Tang, Junjie Wen, Minjie Zhu, Xiaoyu Liu, Chengmeng Li, Ran Cheng, Yaxin Peng, Yan Peng, et al. Coa-vla: Improving vision-language-action models via visual-text chain-of-affordance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[159] Qize Yu, Jiadi You, Yuran Wang, Jiaqi Liang, Bowen Ping, Yang Tian, Yue Chen, Minghong Cai, Zeying Gong, Ruihai Wu, et al. Affordancevla: A vision-language-action model empowering action generation through affordance-aware understanding. *arXiv preprint arXiv:2606.06155*, 2026.

[160] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction in robotics. In *8th Annual Conference on Robot Learning*.

[161] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

[162] Ruisen Tu, Arth Shukla, Sohyun Yoo, Xuanlin Li, Junxi Li, Jianwen Xie, Hao Su, and Zhuowen Tu. Sg-vla: Learning spatially-grounded vision-language-action models for mobile manipulation. *arXiv preprint arXiv:2603.22760*, 2026.

[163] AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, 2018.

[164] Shentong Mo and Paul Pu Liang. Multimed: Massively multimodal and multitask medical understanding. *arXiv preprint arXiv:2408.12682*, 2024.

[165] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[166] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[167] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pages 785–799. PMLR, 2023.

[168] Xukun Li, Yu Sun, Lei Zhang, Bosheng Huang, Yibo Peng, Yuan Meng, Haojun Jiang, Shaoxuan Xie, Guocai Yao, Alois Knoll, et al. Deco: Decoupled multimodal diffusion transformer for bimanual dexterous manipulation with a plugin tactile adapter. *arXiv preprint arXiv:2602.05513*, 2026.

[169] Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International conference on machine learning*, pages 5547–5569. PMLR, 2022.

[170] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[171] Yunxin Li, Shenyuan Jiang, Baotian Hu, Longyue Wang, Wanqi Zhong, Wenhan Luo, Lin Ma, and Min Zhang. Uni-moe: Scaling unified multimodal llms with mixture of experts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[172] John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*, 2017.

[173] Haonan Chen, Jiaming Xu, Hongyu Chen, Kaiwen Hong, Binghao Huang, Chaoqi Liu, Jiayuan Mao, Yunzhu Li, Yilun Du, and Katherine Driggs-Campbell. Multi-modal manipulation via multi-modal policy consensus. *arXiv preprint arXiv:2509.23468*, 2025.

[174] Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8238–8247, 2022.

[175] Yake Wei, Di Hu, Henghui Du, and Ji-Rong Wen. On-the-fly modulation for balanced multimodal learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(1):469–485, 2024.

[176] Bohan Hou, Gen Li, Jindou Jia, Tuo An, Xinying Guo, Sicong Leng, Haoran Geng, Yanjie Ze, Tatsuya Harada, Philip Torr, et al. World model for robot learning: A comprehensive survey. *arXiv preprint arXiv:2605.00080*, 2026.

[177] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

[178] An Dang, Jayjun Lee, Mustafa Mukadam, X. Alice Wu, Bernadette Bucher, Manikantan Nambi, and Nima Fazeli. HydroShear: Hydroelastic shear simulation for tactile sim-to-real reinforcement learning, 2026. URL http://arxiv.org/abs/2603.00446.

[179] Zhengrong Xue, Han Zhang, Jingwen Cheng, Zhengmao He, Yuanchen Ju, Changyi Lin, Gu Zhang, and Huazhe Xu. ArrayBot: Reinforcement learning for generalizable distributed manipulation through touch, 2025. URL http://arxiv.org/abs/2306.16857.

[180] Wenbin Hu, Bidan Huang, Wang Wei Lee, Sicheng Yang, Yu Zheng, and Zhibin Li. Dexterous in-hand manipulation of slender cylindrical objects through deep reinforcement learning with tactile sensing, 2023. URL http://arxiv.org/abs/2304.05141.

[181] Jose A. Barreiros, Aykut Özgün Önol, Mengchao Zhang, Sam Creasey, Aimee Goncalves, Andrew Beaulieu, Aditya Bhat, Kate M. Tsui, and Alex Alspach. Learning contact-rich whole-body manipulation with example-guided reinforcement learning. *Science Robotics*, 10(105):eads6790, 2025. doi: 10.1126/scirobotics.ads6790. URL https://www-science-org.remotexs.ntu.edu.sg/doi/10.1126/scirobotics.ads6790. Publisher: American Association for the Advancement of Science.

[182] Entong Su, Chengzhe Jia, Yuzhe Qin, Wenxuan Zhou, Annabella Macaluso, Binghao Huang, and Xiaolong Wang. Sim2real manipulation on unknown objects with tactile-based reinforcement learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9234–9241, 2024. doi: 10.1109/ICRA57147.2024.10611113. URL https://ieeexplore.ieee.org/abstract/document/10611113.

[183] Yongqiang Zhao, Xuyang Zhang, Zhuo Chen, Matteo Leonetti, Emmanouil Spyrakos-Papastavridis, and Shan Luo. Visual-tactile peg-in-hole assembly learning from peg-out-of-hole disassembly. *IEEE Robotics and Automation Letters*, 11(6):6712–6719, 2026. ISSN 2377-3766. doi: 10.1109/LRA.2026.3679227. URL https://ieeexplore.ieee.org/abstract/document/11458772.

[184] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[185] Haoran Yuan, Weigang Yi, Zhenyu Zhang, Wendi Chen, Yuchen Mo, Jiashi Yin, Xinzhuo Li, Xiangyu Zeng, Chuan Wen, Cewu Lu, Katherine Driggs-Campbell, and Ismini Lourentzou. VTAM: Video-tactile-action models for complex physical interaction beyond VLAs, 2026. URL https://arxiv.org/abs/2603.23481v1.

[186] Tianyu Li, Yihan Li, Zizhe Zhang, and Nadia Figueroa. Flow with the force field: Learning 3d compliant flow matching policies from force and demonstration-guided simulation data, 2026. URL http://arxiv.org/abs/2510.02738.

[187] Noah Geiger, Tamim Asfour, Neville Hogan, and Johannes Lachner. Diffusion-based impedance learning for contact-rich manipulation tasks, 2026. URL http://arxiv.org/abs/2509.19696.

[188] Mingxin Wang, Zhirun Yue, Renhao Lu, Yizhe Li, Zihan Wang, Guoping Pan, Kangkang Dong, Jun Cheng, Yi Cheng, and Houde Liu. PhaForce: Phase-scheduled visual-force policy learning with slow planning and fast correction for contact-rich manipulation, 2026. URL http://arxiv.org/abs/2603.08342.

[189] Zhaokun Yue, Ling Tong, and Kun Qian. PoCoDP3: Pose- and contact-aware visual-tactile policy for contact-rich 3d manipulation. *IEEE Robotics and Automation Letters*, 11(3):2434–2441, 2026. ISSN 2377-3766. doi: 10.1109/LRA.2026.3653314. URL https://ieeexplore.ieee.org/document/11345942/.

[190] Teng Xue, Alberto Rigo, Bingjian Huang, Jiayi Shen, Zhengtong Xu, Nick Colonnese, and Amirhossein H. Memar. Tube diffusion policy: Reactive visual-tactile policy learning for contact-rich manipulation, 2026. URL http://arxiv.org/abs/2604.23609.

[191] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[192] Guo Ye, Zexi Zhang, Xu Zhao, Shang Wu, Haoran Lu, Shihan Lu, and Han Liu. Learning to feel the future: DreamTacVLA for contact-rich manipulation, 2026. URL http://arxiv.org/abs/2512.23864.

[193] Haobo Kang, Hongjun Ma, and Weichang Li. CATCH-FORM-ACTer: Compliance-aware tactile control and hybrid deformation regulation-based action transformer for viscoelastic object manipulation. *IEEE Access*, 13:131998–132005, 2025. ISSN 2169-3536. doi: 10.1109/ACCESS.2025.3591499. URL https://ieeexplore.ieee.org/abstract/document/11088097.

[194] Takumi Kobayashi, Masato Kobayashi, Thanpimon Buamanee, and Yuki Uranishi. Bi-LAT: Bilateral control-based imitation learning via natural language and action chunking with transformers. In *2025 34th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 1609–1616, 2025. doi: 10.1109/RO-MAN63969.2025.11217852. URL https://ieeexplore.ieee.org/abstract/document/11217852. ISSN: 1944-9437.

[195] Pedro Miguel Uriguen Eljuri, Hironobu Shibata, Maeyama Katsuyoshi, Yuanyuan Jia, and Tadahiro Taniguchi. Haptic-informed ACT with a soft gripper and recovery-informed training for pseudo oocyte manipulation, 2025. URL https://ui.adsabs.harvard.edu/abs/2025arXiv250618212U. ADS Bibcode: 2025arXiv250618212U.

[196] Masato Kobayashi, Thanpimon Buamanee, and Takumi Kobayashi. ALPHA- α and bi-ACT are all you need: Importance of position and force information/ control for imitation learning of unimanual and bimanual robotic manipulation with low-cost system. *IEEE Access*, 13:29886–29899, 2025. ISSN 2169-3536. doi: 10.1109/ACCESS.2025.3541200. URL https://ieeexplore.ieee.org/abstract/document/10883984.

[197] Ryo Watanabe, Maxime Alvarez, Pablo Ferreiro, Pavel Savkin, and Genki Sano. FTACT: Force torque aware action chunking transformer for pick-and-reorient bottle task, 2025. URL http://arxiv.org/abs/2509.23112.

[198] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[199] Ruiteng Zhao, Wenshuo Wang, Yicheng Ma, Xiaocong Li, Francis E. H. Tay, Marcelo H. Ang, and Haiyue Zhu. FD-VLA: Force-distilled vision-language-action model for contact-rich manipulation, 2026. URL http://arxiv.org/abs/2602.02142.

[200] Yao Li, Peiyuan Tang, Wuyang Zhang, Chengyang Zhu, Yifan Duan, Weikai Shi, Xiaodong Zhang, Zijiang Yang, Jianmin Ji, and Yanyong Zhang. FAVLA: A force-adaptive fast-slow VLA model for contact-rich robotic manipulation, 2026. URL http://arxiv.org/abs/2602.23648.

[201] Charlotte Morissette, Amin Abyaneh, Wei-Di Chang, Anas Houssaini, David Meger, Hsiu-Chin Lin, Jonathan Tremblay, and Gregory Dudek. Tactile modality fusion for vision-language-action models, 2026. URL http://arxiv.org/abs/2603.14604.

[202] Jimin Lee, Huiwon Jang, Myungkyu Koo, Jungwoo Park, and Jinwoo Shin. Modular sensory stream for integrating physical feedback in vision-language-action models, 2026. URL http://arxiv.org/abs/2604.23272.

[203] Yifei Yang, Anzhe Chen, Zhenjie Zhu, Kechun Xu, Yunxuan Mao, Yufei Wei, Lu Chen, Rong Xiong, and Yue Wang. Direction matters: Learning force direction enables sim-to-real contact-rich manipulation, 2026. URL http://arxiv.org/abs/2602.14174.

[204] Thomas TCK Zhang, Daniel Pfrommer, Chaoyi Pan, Nikolai Matni, and Max Simchowitz. Action chunking and data augmentation yield exponential improvements in behavior cloning for continuous spaces. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=jiWXDvw1Lf.

[205] Yuejiang Liu, Jubayer Ibn Hamid, Annie Xie, Yoonho Lee, Maximilian Du, and Chelsea Finn. Bidirectional decoding: Improving action chunking via guided test-time sampling. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/0d78dd998f7b9ac79604d47a2d79bb0d-Abstract-Conference.html.

[206] Kevin Black, Manuel Y. Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://openreview.net/forum?id=UkR2zO5uww.

[207] Steven Oh, Jason Jingzhou Liu, Tony Tao, Philip Han, Kenneth Shaw, Satoshi Funabashi, Ruslan Salakhutdinov, and Deepak Pathak. Factr 2: Learning external force sensing for commodity robot arms improves policy learning. *arXiv preprint arXiv:2606.12406*, 2026.

[208] Chenrui Tie, Shengxiang Sun, Jinxuan Zhu, Yiwei Liu, Jingxiang Guo, Yue Hu, Haonan Chen, Junting Chen, Ruihai Wu, and Lin Shao. Manual2skill: Learning to read manuals and acquire robotic skills for furniture assembly using vision-language models. *arXiv preprint arXiv:2502.10090*, 2025.

[209] Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024.

[210] Jianke Zhang, Yanjiang Guo, Xiaoyu Chen, Yen-Jen Wang, Yucheng Hu, Chengming Shi, and Jianyu Chen. Hirt: Enhancing robotic control with hierarchical robot transformers. *arXiv preprint arXiv:2410.05273*, 2024.

[211] Minjie Zhu, Yichen Zhu, Jinming Li, Junjie Wen, Zhiyuan Xu, Zhengping Che, Chaomin Shen, Yaxin Peng, Dong Liu, Feifei Feng, et al. Language-conditioned robotic manipulation with fast and slow thinking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4333–4339. IEEE, 2024.

[212] Ingmar Posner. Robots thinking fast and slow: on dual process theory and metacognition in embodied ai. In *Robotics Retrospectives-Workshop at RSS 2020*, 2020.

[213] Muzhen Cai, Xiubo Chen, Yining An, Jiaxin Zhang, Xuesong Wang, Wang Xu, Weinan Zhang, and Ting Liu. Cookbench: A long-horizon embodied planning benchmark for complex cooking scenarios. *arXiv preprint arXiv:2508.03232*, 2025.

[214] Xiao Wang, Lu Dong, Jingchen Sun, Ifeoma Nwogu, Srirangaraj Setlur, and Venu Govindaraju. Mistypilot: An agentic fast-slow thinking llm framework for misty social robots. *arXiv preprint arXiv:2603.03640*, 2026.

[215] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

[216] Anurag Ajay, Seungwook Han, Yilun Du, Shuang Li, Abhi Gupta, Tommi Jaakkola, Josh Tenenbaum, Leslie Kaelbling, Akash Srivastava, and Pulkit Agrawal. Compositional foundation models for hierarchical planning. *Advances in Neural Information Processing Systems*, 36:22304–22325, 2023.

[217] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[218] Qingwen Bu, Hongyang Li, Li Chen, Jisong Cai, Jia Zeng, Heming Cui, Maoqing Yao, and Yu Qiao. Towards synergistic, generalized, and efficient dual-system for robotic manipulation. *arXiv preprint arXiv:2410.08001*, 2024.

[219] Hao Chen, Jiaming Liu, Chenyang Gu, Zhuoyang Liu, Renrui Zhang, Xiaoqi Li, Xiao He, Yandong Guo, Chi-Wing Fu, Shanghang Zhang, et al. Fast-in-slow: A dual-system vla model unifying fast manipulation within slow reasoning. *Advances in Neural Information Processing Systems*, 38:98049–98083, 2026.

[220] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

[221] Tony J Prescott and Stuart P Wilson. Understanding brain functional architecture through robotics. *Science Robotics*, 8(78):eadg6014, 2023.

[222] John J Craig. Introduction to robotics. 2005.

[223] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*, volume 2. Wiley New York, 2020.

[224] Marco Laghi, Arash Ajoudani, Manuel G Catalano, and Antonio Bicchi. Unifying bilateral teleoperation and tele-impedance for enhanced user experience. *The International Journal of Robotics Research*, 39(4):514–539, 2020.

[225] Elham Hamid, Nicolas Herzig, Sara-Adela Abad, and Thrishantha Nanayakkara. A state-dependent damping method to reduce collision force and its variability. *IEEE Robotics and Automation Letters*, 6(2):3025–3032, 2021.

[226] Ruize Wang, Peng Cheng, Qi Ye, Jiming Chen, and Gaofeng Li. A two-stage dynamic parameters identification approach based on kalman filter for haptic rendering in telerobotics. *IEEE Robotics and Automation Letters*, 10 (7):6600–6607, 2025. doi: 10.1109/LRA.2025.3568613.

[227] Chaoyi Zhang, Shiyang Liu, Chao Zeng, Yiming Jiang, and Jing Luo. A robot humanoid control framework through human arm active endpoint stiffness and direction adaptive compensation. *IEEE Robotics and Automation Letters*, 9(9):7557–7564, 2024.

[228] Yiming Chen, Yuhao Zhang, Xingwei Zhao, Qiang Xie, Kun Yang, Bo Tao, and Han Ding. Physical human–robot interaction based on adaptive impedance control for robotic-assisted total hip arthroplasty. *IEEE/ASME Transactions on Mechatronics*, 29(6):4674–4686, 2024.

[229] Junling Fu, Ilaria Burzo, Elisa Iovene, Jianzhuang Zhao, Giancarlo Ferrigno, and Elena De Momi. Optimization-based variable impedance control of robotic manipulator for medical contact tasks. *IEEE Transactions on Instrumentation and Measurement*, 73:1–8, 2024.

[230] Fanny Ficuciello, Luigi Villani, and Bruno Siciliano. Variable impedance control of redundant manipulators for intuitive human–robot physical interaction. *IEEE Transactions on Robotics*, 31(4):850–863, 2015.

[231] Chenwei Gong, Fei Zhao, Zhiwei Liao, Tao Tao, Xiao Wang, and Xuesong Mei. Simple-rotation angle/axis representations based second-order impedance control. *IEEE Robotics and Automation Letters*, 9(6):5831–5838, 2024. doi: 10.1109/LRA.2024.3396090.

[232] Christian Ott, Ranjan Mukherjee, and Yoshihiko Nakamura. Unified impedance and admittance control. In *2010 IEEE international conference on robotics and automation*, pages 554–561. IEEE, 2010.

[233] Sami Haddadin and Erfan Shahriari. Unified force-impedance control. *The International Journal of Robotics Research*, 43(13):2112–2141, 2024.

[234] Jingdong Chen and Paul I Ro. Human intention-oriented variable admittance control with power envelope regulation in physical human-robot interaction. *Mechatronics*, 84:102802, 2022.

[235] Lu Chen, Lipeng Chen, Xiangchi Chen, Haojian Lu, Yu Zheng, Jun Wu, Yue Wang, Zhengyou Zhang, and Rong Xiong. Compliance while resisting: A shear-thickening fluid controller for physical human-robot interaction. *The International Journal of Robotics Research*, 43(11):1731–1769, 2024.

[236] H. Seraji. Adaptive admittance control: an approach to explicit force control in compliant motion. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2705–2712 vol.4, 1994. doi: 10.1109/ROBOT.1994.350927.

[237] Gitae Kang, Hyun Seok Oh, Joon Kyue Seo, Uikyum Kim, and Hyouk Ryeol Choi. Variable admittance control of robot manipulators based on human intention. *IEEE/ASME Transactions on Mechatronics*, 24(3):1023–1032, 2019.

[238] Federica Ferraguti, Chiara Talignani Landi, Lorenzo Sabattini, Marcello Bonfe, Cesare Fantuzzi, and Cristian Secchi. A variable admittance control strategy for stable physical human–robot interaction. *The International Journal of Robotics Research*, 38(6):747–765, 2019.

[239] Hsieh-Yu Li, Ishara Paranawithana, Liangjing Yang, Terence Sey Kiat Lim, Shaohui Foong, Foo Cheong Ng, and U-Xuan Tan. Stable and compliant motion of physical human–robot interaction coupled with a moving environment using variable admittance and adaptive control. *IEEE Robotics and Automation Letters*, 3(3): 2493–2500, 2018.

[240] Tsuneo Yoshikawa, Toshiharu Sugie, and Masaki Tanaka. Dynamic hybrid position/force control of robot manipulators-controller design and experiment. *IEEE Journal on Robotics and Automation*, 4(6):699–705, 2002.

[241] Tsuneo Yoshikawa. Dynamic hybrid position/force control of robot manipulators–description of hand constraints and calculation of joint driving force. *IEEE Journal on Robotics and Automation*, 3(5):386–392, 2003.

[242] William D Fisher and M Shahid Mujtaba. Hybrid position/force control: A correct formulation. *The International journal of robotics research*, 11(4):299–311, 1992.

[243] Mehrzad Namvar and Farhad Aghili. Adaptive force-motion control of coordinated robots interacting with geometrically unknown environments. *IEEE Transactions on Robotics*, 21(4):678–694, 2005.

[244] Stefan Scherzinger, Arne Roennau, and Rüdiger Dillmann. Forward dynamics compliance control (fdcc): A new approach to cartesian compliance for robotic manipulators. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4568–4575, 2017. doi: 10.1109/IROS.2017.8206325.

[245] Davide Liconti, Yuning Zhou, Yasunori Toshimitsu, Ronan Hinchet, and Robert K. Katzschmann. A benchmark of dexterity for anthropomorphic robotic hands. *arXiv preprint arXiv:2604.09294*, 2026.

[246] Chengbo Yuan, Zicheng Zhang, Mingjie Zhou, Wendi Chen, Yi Wang, Zhuoyang Liu, Dantong Niu, Shuo Wang, Hui Zhang, Wenkang Zhang, Yingdong Hu, Yuanqing Gong, Wanli Xing, Chuan Wen, Cewu Lu, Kaifeng Zhang, and Yang Gao. Ftp-1: A generalist foundation tactile policy across tactile sensors for contact-rich manipulation. *arXiv preprint arXiv:2606.13102*, 2026.

[247] Yan Zhang, WANG Zheng, Pengpeng Zeng, Xing Xu, Jingkuan Song, and Heng Tao Shen. Cross-tactile sensor representation learning. In *Forty-third International Conference on Machine Learning*.

[248] Youngsun Wi, Jessica Yin, Elvis Xiang, Akash Sharma, Jitendra Malik, Mustafa Mukadam, Nima Fazeli, and Tess Hellebrekers. Tactalign: Human-to-robot policy transfer via tactile alignment. 2025.

[249] Zhengtong Xu, Yeping Wang, Ben Abbatematteo, Jom Preechayasomboon, Sonny Chan, Nick Colonnese, and Amirhossein H Memar. Contact-grounded policy: Dexterous visuotactile policy with generative contact grounding. *arXiv preprint arXiv:2603.05687*, 2026.

[250] Shuai Tian, Yupeng Zheng, Yuhang Zheng, Songen Gu, Yujie Zang, Yuxing Qin, Weize Li, Haoran Li, Wenchao Ding, and Dongbin Zhao. Vt-wam: Visual-tactile world action model for contact-rich manipulation, 2026. URL https://arxiv.org/abs/2607.02503.

[251] Yunfan Lou, Yifan Ye, Yankai Fu, Jun Cen, Xiaowei Chi, Yaoxu Lyu, Peidong Jia, Sirui Han, Zhihe Lu, and Shanghang Zhang. Dream-tac: A unified tactile world action model for contact-rich robot manipulation, 2026. URL https://arxiv.org/abs/2606.08737.

[252] Haotian He, Zeyu Yan, Qipeng Liu, Ning Guo, and Wenzhao Lian. Fawam: Force-aware world action models for closed-loop contact-rich manipulation, 2026. URL https://arxiv.org/abs/2606.08555.

[253] Yujie Zang, Yuhang Zheng, Xian Nie, Yupeng Zheng, Shuai Tian, Songen Gu, Chen Gao, Zining Wang, Shuicheng Yan, and Wenchao Ding. Tacforesight: Force-guided tactile world model for contact-rich manipulation, 2026. URL https://arxiv.org/abs/2606.11184.

[254] Zhiyuan Zhang, Pokuang Zhou, Kaidi Zhang, Adeesh Desai, Temitope Amosa, Davood Soleymanzadeh, Jiuzhou Lei, Minghui Zheng, and Yu She. Contactworld: What matters in vision-tactile world models for contact-rich manipulation, 2026. URL https://arxiv.org/abs/2606.13877.

[255] Siyu Wu, Linjing You, Junjie Zhu, Yaozu Liu, Changhao Zhang, Jian Liu, Weiqiang Wang, Qi Li, Jituo Li, and Hengshuang Zhao. Tactile-wam: Touch-aware world action model with tactile asymmetric attention, 2026. URL https://arxiv.org/abs/2606.26663.

[256] Yilin Wu, Zilin Si, Zeynep Temel, Oliver Kroemer, and Andrea Bajcsy. Inference-time policy steering via vision and touch. *arXiv preprint arXiv:2606.14981*, 2026. doi: 10.48550/arXiv.2606.14981.

[257] Jacky Kwok, Xilun Zhang, Mengdi Xu, Yuejiang Liu, Azalia Mirhoseini, Chelsea Finn, and Marco Pavone. Scaling verification can be more effective than scaling policy learning for vision-language-action alignment. *arXiv preprint arXiv:2602.12281*, 2026. doi: 10.48550/arXiv.2602.12281.

[258] Jacky Kwok, Shulu Li, Pranav Atreya, Yuejiang Liu, Yixing Jiang, Chelsea Finn, Marco Pavone, Ion Stoica, and Azalia Mirhoseini. LLM-as-a-verifier: A general-purpose verification framework. *arXiv preprint arXiv:2607.05391*, 2026. doi: 10.48550/arXiv.2607.05391.

[259] Yuejiang Liu, Fan Feng, Lingjing Kong, Weifeng Lu, Jinzhou Tang, Kun Zhang, Kevin Murphy, Chelsea Finn, and Yilun Du. World Action Verifier: Self-improving world models via forward-inverse asymmetry. *arXiv preprint arXiv:2604.01985*, 2026. doi: 10.48550/arXiv.2604.01985.

[260] Carolina Higuera, Sergio Arnaud, Byron Boots, Mustafa Mukadam, Francois Robert Hogan, and Franziska Meier. Visuo-tactile world models, 2026. URL http://arxiv.org/abs/2602.06001.

[261] Yating Lin, Zixuan Huang, Fan Yang, and Dmitry Berenson. AnoF-Diff: One-step diffusion-based anomaly detection for forceful tool use. *arXiv preprint arXiv:2509.15153*, 2025. doi: 10.48550/arXiv.2509.15153.

[262] Anthony Liang, Yigit Korkmaz, Jiahui Zhang, Minyoung Hwang, Abrar Anwar, Sidhant Kaushik, Aditya Shah, Alex S. Huang, Luke Zettlemoyer, Dieter Fox, Yu Xiang, Anqi Li, Andreea Bobu, Abhishek Gupta, Stephen Tu, Erdem Biyik, and Jesse Zhang. Robometer: Scaling general-purpose robotic reward models via trajectory comparisons. *arXiv preprint arXiv:2603.02115*, 2026. doi: 10.48550/arXiv.2603.02115.

[263] Tony Lee, Andrew Wagenmaker, Karl Pertsch, Percy Liang, Sergey Levine, and Chelsea Finn. RoboReward: General-purpose vision-language reward models for robotics. *arXiv preprint arXiv:2601.00675*, 2026. doi: 10.48550/arXiv.2601.00675.

[264] M. Sonoda, R. Hinchet, A. Kazemipour, Y. Toshimitsu, and Robert K. Katzschmann. A sensorless, inherently compliant anthropomorphic musculoskeletal hand driven by electrohydraulic actuators, 2026. URL https://arxiv.org/abs/2603.24357.

[265] O. S. Neumann, L. S. Ewering, and Robert K. Katzschmann. Integrated by design: A hybrid robotic palm fusing a functionally-dense compliant matrix with an actuated skeleton. *IEEE Robotics and Automation Letters*, 2026.

[266] A. Kazemipour and Robert K. Katzschmann. Decoupling torque and stiffness: A unified modeling and control framework for antagonistic artificial muscles. In *2026 IEEE 9th International Conference on Soft Robotics (RoboSoft)*, pages 813–820, 2026.