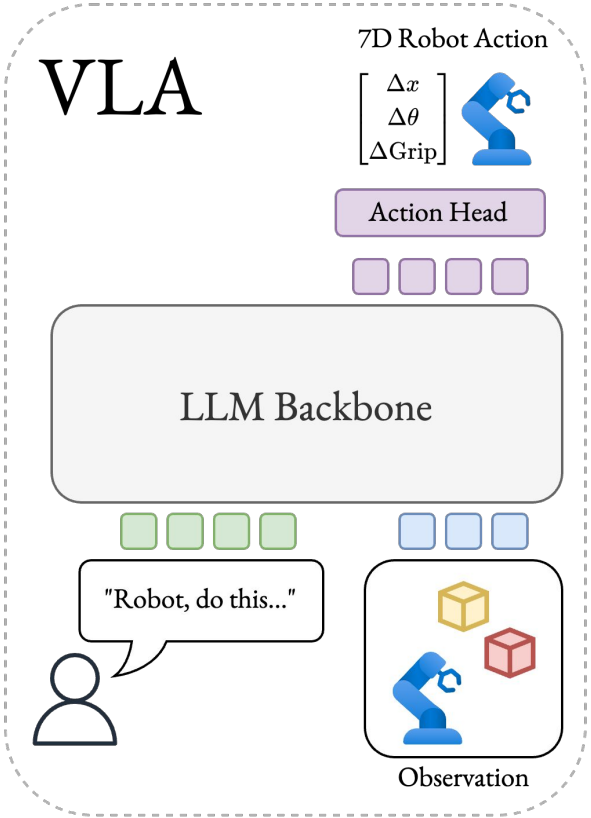
# Interpretable Vision-Language-Action Models via Skill Conditioning

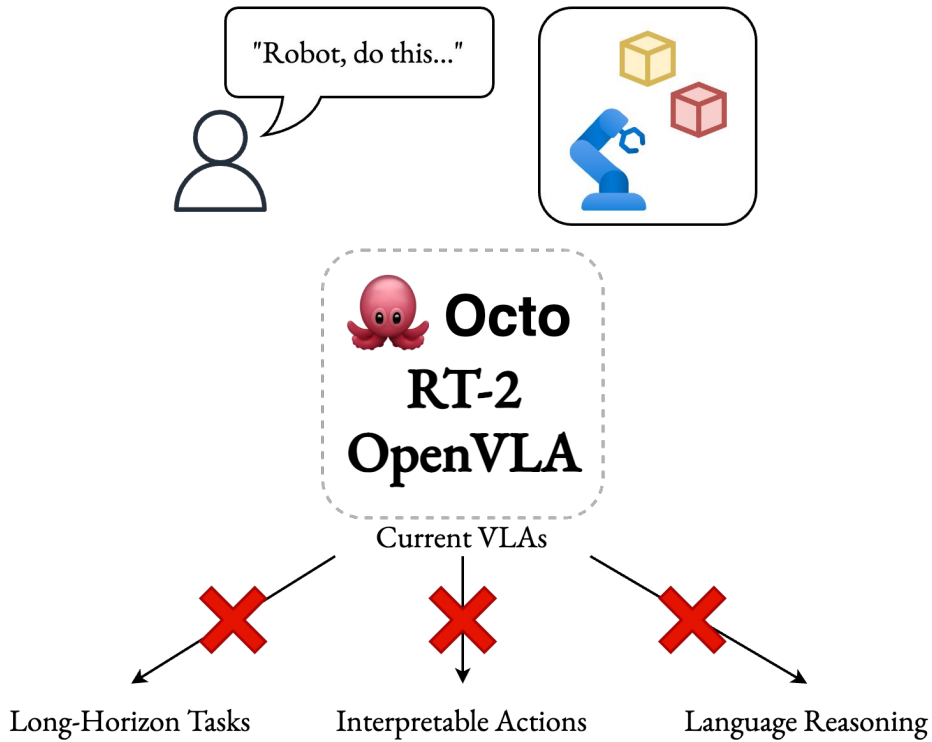
**Brandon Y. Yang*** **Yen-Ling Kuo***

*University of Virginia

{branyang, ylkuo}@virginia.edu

# Vision-Language-Action (VLA) Models

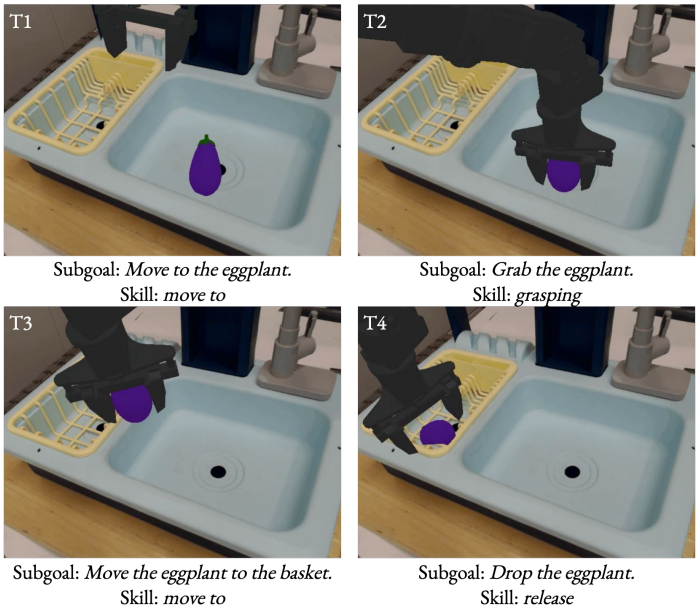


# **SkillVLA** (**Skill**-driven **V**ision-**L**anguage-**A**ction Model)

**SkillVLA** improves *long-horizon language-conditioned robotic policies* and VLA *interpretability* by **grounding** action outputs with synthesized ***subgoal instructions*** and a learned ***skill library***.

**Task Instruction:** ***put eggplant in basket***



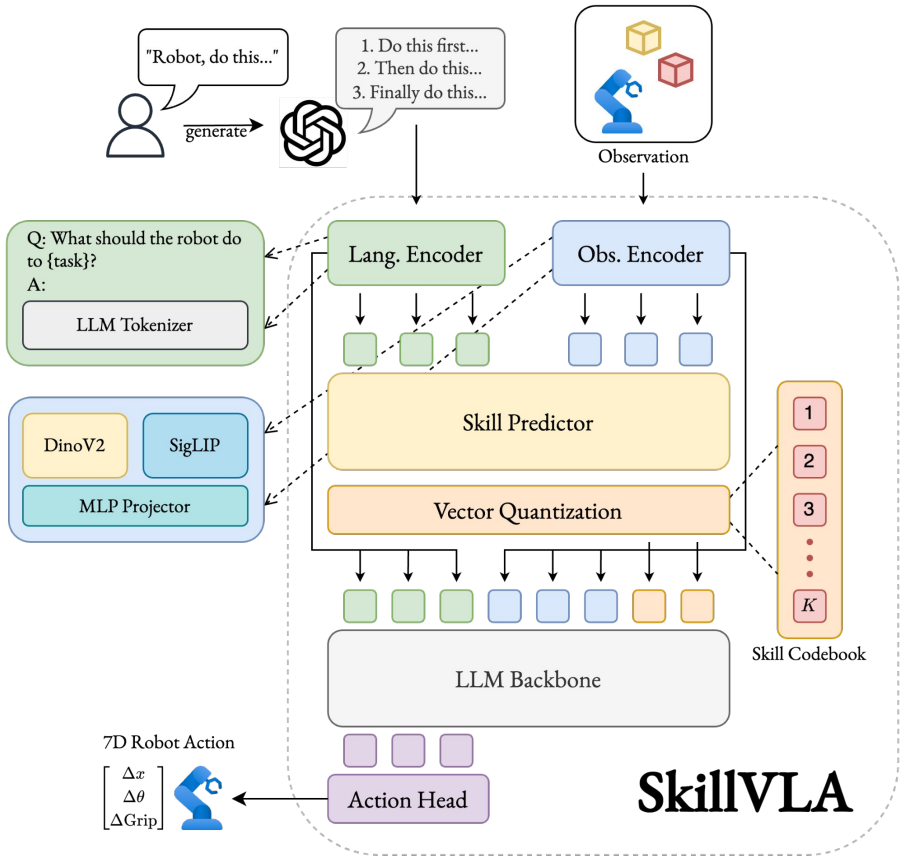

Subgoal: *Move to the eggplant.*
Skill: *move to*

Subgoal: *Grab the eggplant.*
Skill: *grasping*

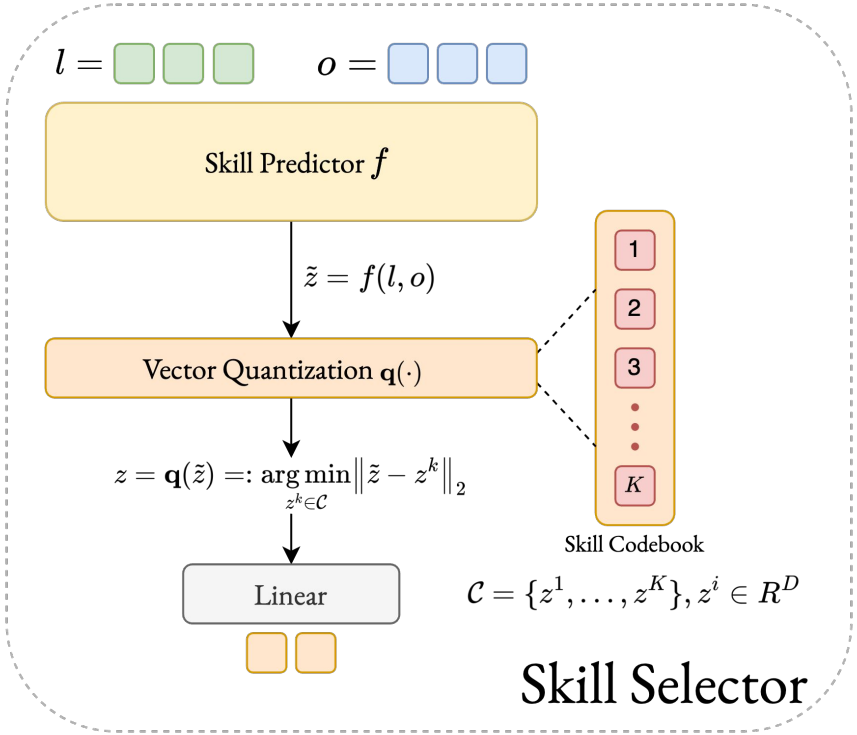
Subgoal: *Move the eggplant to the basket.*
Skill: *move to*

Subgoal: *Drop the eggplant.*
Skill: *release*

Skills are latent variables, meaning we don't have textual captions for them. The demo illustrates how incorporating skill enhances interpretability.



# Skill Predictor and Vector Quantization




Skill Predictor:

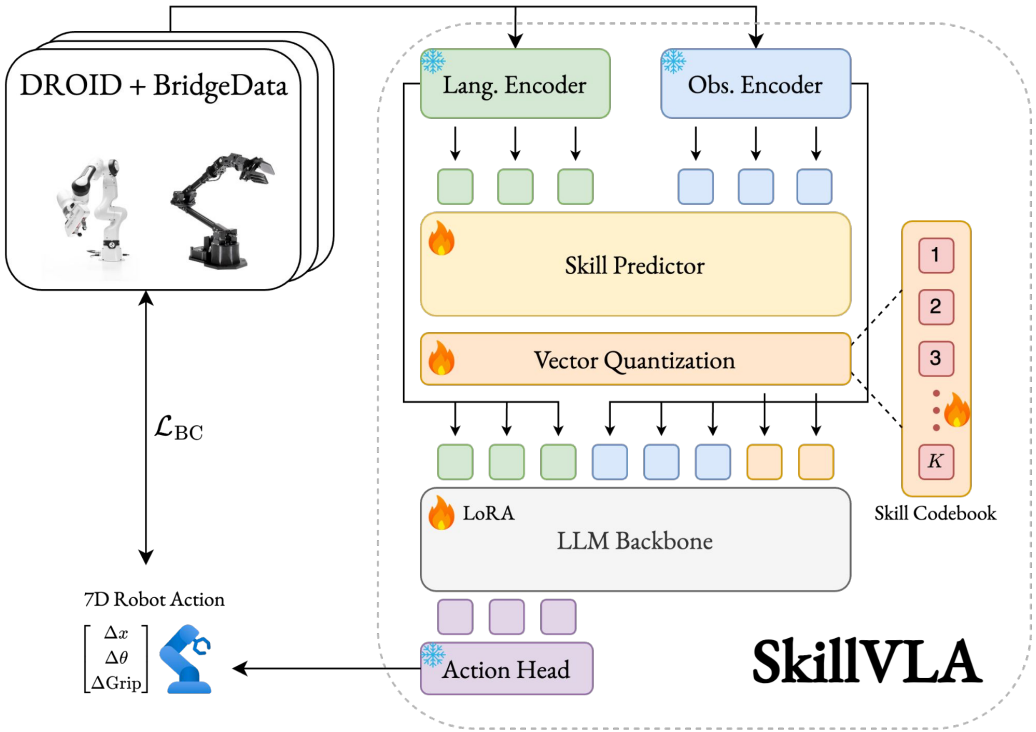
- Receives language and observation as input
- Outputs a *skill code*
- Implemented as either Causal Transformer or MLP

Vector Quantization (VQ):

- Receives a *skill code*
- Outputs the closest *Codebook* entry from the *skill code*
- Codebook trained End-to-end

# **SkillVLA** Training

- Load Pretrained weights from OpenVLA:
  - 🏋️ Language Encoder
  - 🏋️ Observation Encoder
  - 🏋️ LLM Backbone
- Freeze:
  - ❄️ Language Encoder
  - ❄️ Observation Encoder
- Train:
  - 🔥 Skill Selector components
- Finetune:
  - 🔥 LoRA LLM Backbone

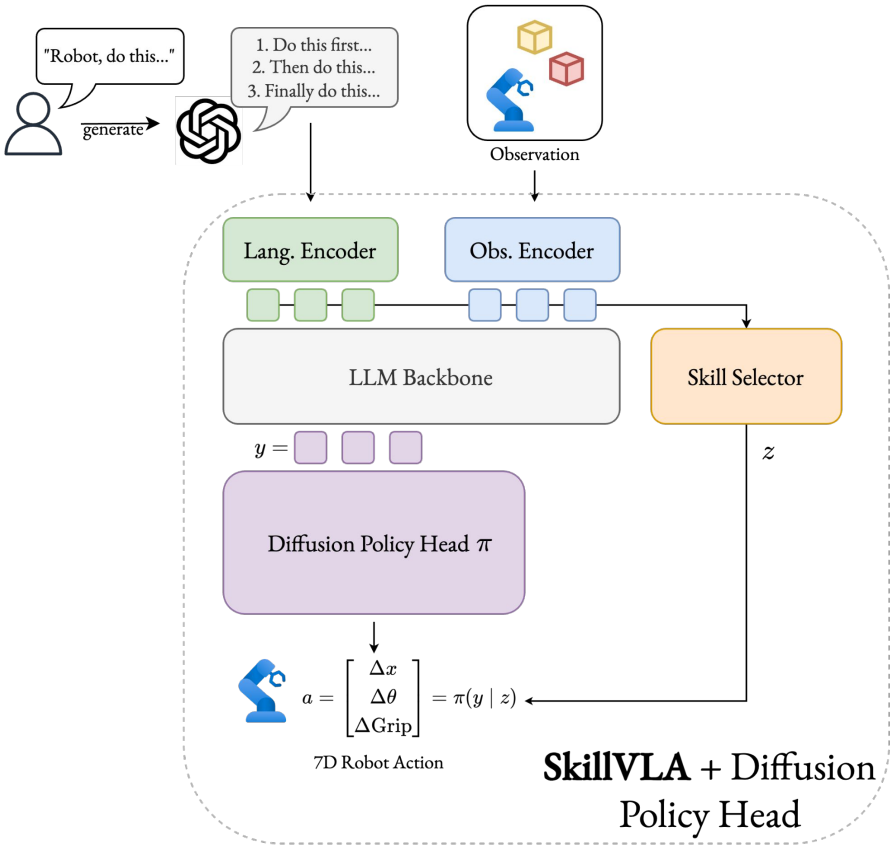


# Preliminary Results

| WidowX+Bridge Evaluation Setup | Policy | Put Spoon on Towel | |
|---|---|---|---|
| | | Grasp Spoon | Success |
| SIMPLER Eval (Visual Matching) | OpenVLA | 0.041 | 0.000 |
| SIMPLER Eval (Visual Matching) | **SkillVLA** | **0.270** | **0.030** |

Table 1: Performance comparison between OpenVLA and SkillVLA on the task of putting a spoon on a towel under the SIMPLER Eval (Visual Matching) setup. We report final success rate ("Success") as well as partial success rate ("Grasp Spoon").

# Future work: **SkillVLA** + Diffusion Policy Head




Use Diffusion Policy *conditioned* on selected skill to generate action.

Pros: More clear that the predicted action is *grounded* on the selected skill.
Cons: Not trivial to implement, and need additional loss functions.