Frequency Reparameterization and Anchored Residuals for Flow Matching Robot Policies

Alan Zhao sxuan@stanford.edu **Juhyun Jung** jj0523@stanford.edu
Stanford University | CS224R Spring 2026 | June 8, 2026

Extended Abstract

Context. Generative imitation policies — Diffusion Policy and Flow Matching (FM) variants — predict short multi-step *action chunks* and execute the first T_a steps in a receding-horizon loop. The chunk is almost universally represented in the raw time domain. We investigate whether a temporal *frequency* representation — per-chunk sparse Discrete Cosine Transform (DCT) — can be a better learning target, and whether the bins removed by DCT truncation can be added back through a small learned residual model.

Research questions. We evaluate four questions on four manipulation benchmarks (Robomimic *Can*/*Square*, *Push-T*, ManiSkill *PegInsertionSide*): **(Q1)** does changing the action target from raw time-domain chunks to DCT/Sparse-DCT improve contact-rich tasks; **(Q2)** what retained bandwidth K works best under Sparse-DCT compression; **(Q3)** is the gain due to frequency reparameterization or compression; **(Q4)** can an anchored DCT-residual recover the bins discarded by truncation? **Key empirical findings.** No single representation wins on all four tasks. Full-Frequency DCT improves mean success over Raw FM from 0.600 to 0.655, showing that the basis change alone is helpful on average. The best fixed global Sparse-DCT bandwidth (K=8 or 12) reaches 0.660 (+10% relative to Raw FM), while a *post-hoc/oracle* task-selected K ({Can 8, Square 4, Push-T 12, Peg 8}) reaches 0.715 (+19.2% relative). Top-K DCT energy is $> 97\%$ for every $K \in \{4, 8, 12\}$, yet same-K success differs by up to **19 points** between Low-Freq and per-chunk Sparse selection — energy is *not* a K-predictor. Relative to the stochastic base-only protocol, a gated DCT residual at λ=0.25 slightly improves Can (0.98→1.00), hurts Square (0.70→0.60), leaves Push-T unchanged (0.72→0.72), hurts PegInsertionSide (0.42→0.36), and lowers the mean from 0.705 to 0.670.

Method	Can	Square	Push-T	Peg	Mean
Raw FM	0.96	0.24	**0.86**	0.34	0.600
DCT Full-Freq	0.98	0.60	0.68	0.36	0.655
Post-hoc Sparse-DCT ($K^\star$)	0.98	0.68	0.84	0.36	**0.715**
Stochastic base only (λ=0)	0.98	**0.70**	0.72	**0.42**	0.705
Anchored Residual (λ=0.25)	**1.00**	0.60	0.72	0.36	0.670

Closed-loop success rate, 50 rollouts, seed 2. ***Bold*** *= per-column winner.*

Mechanism. A direct spectral analysis of expert demonstrations (§6.2) shows all four tasks are strongly low-frequency: the first K=4 DCT coefficients retain $\geq 97\%$ of energy, with FFT and Morlet decompositions in agreement. Three of four tasks (Can, Square, Peg) fit a *spectrum-decay-predicts-K* hypothesis within ±2 bins of their reconstruction knee. Push-T is a clean counterexample: most compressible expert spectrum (99.7% at K=4) yet largest optimal closed-loop K (K=12) — implying K controls more than reconstruction fidelity; policy expressivity and out-of-distribution recovery also contribute. Per-channel breakdown reveals rotation channels carry essentially no high-frequency energy (ratio ≈ 0.001 above K=4), motivating per-channel rather than global K. **Contributions.** (1) A controlled comparison of Raw FM, Full-Frequency DCT, and Sparse-DCT targets, showing that frequency reparameterization alone improves average success but does not dominate every task. (2) A per-task K sweep showing no universal compression rule. (3) A failure-mode result decoupling energy retention from closed-loop success at fixed chunking. (4) A stochastic-base residual evaluation showing that the residual does not improve average closed-loop success over the base it corrects.

Code: https://github.com/SXuanAlan/cs224r_project

1 Abstract

Generative imitation policies represented in the raw time domain may discard a useful inductive bias: action chunks are smooth and concentrated at low temporal frequencies. We study per-chunk sparse Discrete Cosine Transform (DCT) as an alternative action target for Flow Matching policies, and whether the bins removed by DCT truncation can be recovered through a learned residual model. On four manipulation benchmarks (Robomimic Can / Square, Push-T, ManiSkill PegInsertionSide), DCT Full-Freq improves mean closed-loop success over Raw FM from 0.600 to 0.655, the best fixed global Sparse-DCT bandwidth reaches 0.660, and post-hoc/oracle task-selected bandwidth reaches 0.715, but no single K dominates. A spectral analysis of expert demonstrations confirms strong low-frequency concentration ($\geq 97\%$ of energy in the first K=4 coefficients on every task) and reveals a per-channel asymmetry: rotation has a near-zero high-frequency tail, while translation and gripper carry the small high-frequency content. Under the stochastic-base protocol, disabling the residual (λ=0) is stronger on average than injecting it at λ=0.25 (0.705$\rightarrow$0.670): the residual slightly improves Can, hurts Square and PegInsertionSide, and leaves Push-T unchanged. Our findings argue that representation selection is a more reliable lever than residual gating in this regime, and that per-channel rather than global bandwidth selection is the most promising next step.

2 Introduction

Recent generative imitation policies — Diffusion Policy [1] and Flow Matching (FM) [3, 4] variants — predict short *action chunks* of length H rather than single actions, and execute the first $T_a \leq H$ steps in a receding-horizon loop. Chunking reduces compounding error and stabilizes contact-rich manipulation. Almost universally, however, the chunk is represented and learned in the raw time domain.

A small but growing line of work has begun to ask whether a different *basis* for the chunk could be a better learning target. FAST [5] tokenizes action sequences with per-chunk sparse DCT for vision–language–action models, citing fast convergence and compact representations. ACT [2] and Diffusion Policy both keep the time-domain basis but ensemble across overlapping chunk predictions to suppress jitter. HYDRA [6] argues for task-dependent action representations but in non-generative behavioral cloning. None of these provide a controlled study of (a) how much of the chunk's temporal bandwidth should be retained, or (b) whether the discarded high-frequency content can be added back through a learned residual.

We supply that controlled study. Concretely, we ask:

Q1. Does changing the action target from raw time-domain chunks to DCT/Sparse-DCT improve contact-rich tasks?

Q2. Under matched Sparse-DCT compression, what retained bandwidth K works best, and is there a universal K across tasks?

Q3. Is the gain due to frequency reparameterization or compression? We separate the basis-change effect using DCT Full-Freq from the truncation effect using Sparse-DCT $K \in \{4, 8, 12\}$.

Q4. Can a residual model placed around a frozen Sparse-DCT base recover the high-frequency components that truncation discards?

Contributions. (1) A controlled comparison of Raw FM, DCT Full-Freq, and Sparse-DCT targets, showing that frequency reparameterization alone improves average success but does not dominate every task. (2) A per-task K sweep showing no universal compression rule: fixed global K=8/12 reaches 0.660, while post-hoc/oracle task-selected K reaches 0.715. (3) A failure-mode result that energy retention is decoupled from closed-loop success at fixed chunking, with same-K Low-Freq vs. Sparse gaps up to 19 points. (4) A stochastic-base residual evaluation showing that the residual does not improve average closed-loop success over the base it corrects.

3 Related Work

Action-chunked imitation learning. Action Chunking with Transformers (ACT) [2] generates sequences of future actions and executes them under temporal ensembling to reduce compounding error for fine-grained bimanual manipulation. Diffusion Policy [1] models the conditional distribution over action sequences with a diffusion model and executes chunks in a receding-horizon loop. Both fix the action representation to the raw time domain. Our work keeps the chunked, receding-horizon setting but treats the chunk's *basis* as a design variable.

Flow Matching policies. Flow Matching [3, 4] trains a continuous-time vector field from noise to data without simulating SDE rollouts during training, and has been adopted for visuomotor policy learning because it matches

Diffusion Policy quality at lower sampling cost. We use a Temporal U-Net Flow Matching backbone identical across all method variants in this study, so any difference comes from the action target.

Frequency-domain action representations. FAST [5] uses per-chunk DCT plus learned tokenization for VLA models, motivated by the strong temporal correlation in robot action sequences. FAST reports faster training and more compact sequences, but it does not ablate the retained bandwidth K or test whether the truncated components can be added back. Earlier offline work on signal compression [11] establishes that DCT concentrates correlated low-bandwidth signals into a small number of coefficients — the same property motivates its use for action chunks. Our project is the first controlled study, to our knowledge, of how truncated DCT chunks behave in *closed-loop* generative imitation policies and whether learned residuals can repair them.

Residual policy learning. Residual RL [7] and residual policy learning [8] add a learned correction on top of a fixed controller or prior. Our anchored frequency residual follows the same structural template, but the prior is a learned Sparse-DCT FM policy rather than a hand-engineered controller, and the residual is supervised in the DCT basis through a Flow Matching loss plus a soft frequency gate. HYDRA [6] similarly argues for task-dependent representations, but in non-generative behavioral cloning; we extend the question into Flow Matching chunked policies.

4 Method

4.1 Action chunks and the temporal DCT basis

Let $A \in \mathbb{R}^{H\times d}$ denote an action chunk of horizon H=16 steps and dimensionality d (per-task), and let DCT and IDCT be the orthonormal Type-II Discrete Cosine Transform along the time axis, applied independently per channel: $Z = \mathrm{DCT}(A), \quad A = \mathrm{IDCT}(Z), \quad Z \in \mathbb{R}^{H\times d}$. The DCT concentrates smooth signals into the first few coefficients, so truncating high-frequency bins compresses the chunk in a data-adapted way.

4.2 Sparse-DCT target

For each chunk and each channel we keep the per-chunk top-K magnitude bins:

$$m_k = \begin{cases} 1, & k \in \mathrm{TopK}(|Z|), \\ 0, & \text{otherwise}, \end{cases} \qquad \hat{A} = \mathrm{IDCT}(m \odot Z), \qquad K \in \{4, 8, 12\}. \tag{1}$$

This differs from a fixed low-pass mask in two ways: the kept bins are selected *per chunk*, and they are chosen by magnitude rather than by position. The FM model is trained to predict the sparse coefficients $\hat{Z} = m \odot Z$ from the observation history; at execution time we sample $\hat{Z}$ from the trained vector field and reconstruct $\hat{A} = \mathrm{IDCT}(\hat{Z})$.

4.3 Anchored frequency residual

The Sparse-DCT target throws away bins by construction. We ask whether the discarded content can be added back through a learned residual model around a frozen Sparse-DCT base. In the final experiments, the residual wrapper uses the best per-task Sparse-DCT anchor and samples the base stochastically, matching the evaluation protocol. Let $a_{\mathrm{base}}(o_t)$ be that stochastic base sample and a_{demo} the ground-truth demonstration chunk. The residual model is trained on the DCT of the time-domain residual:

$$z_r = \mathrm{DCT}\big(a_{\mathrm{demo}} - a_{\mathrm{base}}(o_t)\big), \quad z_{\mathrm{train}} = g(o_t) \odot z_r, \tag{2}$$

where $g(o_t) \in [0,1]^{H\times d}$ is a learned soft frequency gate predicted from the observation, initialized small (`init_logit`$= -2.0$). At execution time:

$$\hat{a} = a_{\mathrm{base}}(o_t) + \lambda \cdot \mathrm{IDCT}\big(g(o_t) \odot \hat{z}_r\big), \qquad \lambda \in \{0, 0.25, 0.5, 0.75, 1.0\}. \tag{3}$$

The setting λ=0 disables the residual inside the same wrapper, so it is the stochastic base-only control for Q4 rather than a separate standalone Sparse-DCT checkpoint. The training loss combines the standard FM regression on the gated target with a small reconstruction term in action space (weight 0.1) and an ℓ_1 sparsity prior on the gate (weight 10^{-3}): $\mathcal{L} = \mathcal{L}_{\mathrm{FM}}(z_{\mathrm{train}}) + 0.1\,\|a_{\mathrm{demo}} - \hat{a}\|_2^2 + 10^{-3}\,\|g\|_1$. The reconstruction term is computed with λ_{train}=0.25; at evaluation we reuse the same trained residual and vary only the execution-time λ. The base is frozen throughout, so the residual cannot rewrite the base prediction — it can only add a small, gated correction. The reported λ=0.25 scale is deliberately conservative, and the full sweep checks whether any global residual weight improves over λ=0.

4.4 Why this design choice?

A retired earlier variant (Raw FM as base, high-frequency-only residual) failed broadly: a high-frequency-only residual cannot repair the low- and mid-frequency timing errors of a Raw FM base. We therefore (i) moved the base from Raw FM to Sparse-DCT, which has stronger smooth structure, and (ii) made the residual full-spectrum but gated, so it can in principle borrow energy from any bin where the base happens to be wrong. Anchoring the residual around the best per-task Sparse-DCT isolates the residual question: *does adding back the discarded bins help?*

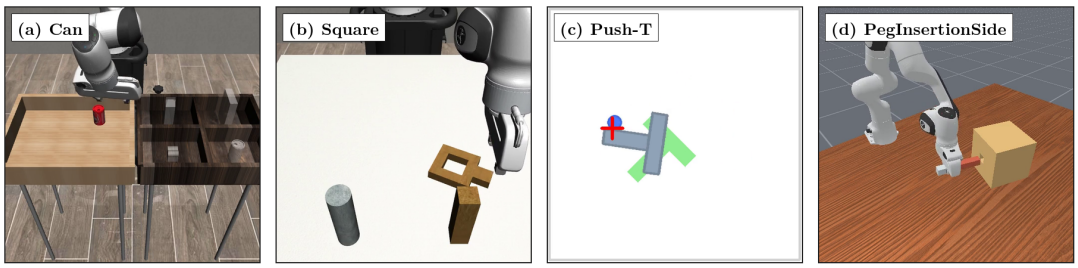

Figure 1: The four manipulation benchmarks used throughout the study. Robomimic **Can** and **Square** (PH split), **Push-T** (state-based), and ManiSkill **PegInsertionSide**. All four use H=16 action chunks, T_a=8 executed actions per step, and the same Temporal U-Net Flow Matching backbone.

5 Experimental Setup

Tasks. We use four manipulation environments selected to span single-arm position-controlled and dual-arm contact-rich behavior at multiple difficulty levels: Robomimic Can and Square (proficient-human (PH) split) [9], Push-T (state-based proprioceptive) [1], and ManiSkill PegInsertionSide [10]. Figure 1 summarizes the suite.

Policy backbone. All variants share the same Temporal U-Net Flow Matching backbone [3, 1] with a learned sinusoidal time embedding, observation horizon of 2 frames, action horizon H=16, and execution horizon T_a=8 (overridden to task-specific defaults where relevant). Training: 300 epochs, up to 80 train batches per epoch, AdamW with lr=1×10^{-4} and weight decay 10^{-6}, EMA on the model weights.

Evaluation. Closed-loop rollouts in each environment, n=50 episodes per checkpoint, seed 2 by default. For sensitivity we additionally evaluate all methods on Robomimic at seeds 0 and 1 from an earlier sweep (Section 6.1). Success is the environment-defined binary success indicator at the end of the episode.

Baselines. (i) *Raw FM* predicts a_{demo} in the time domain. (ii) *DCT Full-Freq* predicts all DCT coefficients without truncation. (iii) *Sparse-DCT* $K \in \{4, 8, 12\}$ predicts the top-K sparse spectrum of Eq. (1). We also report a *post-hoc* $K^\star$ row selected by closed-loop success per task; this is an oracle diagnostic, not an offline deployment rule. (iv) *Low-Freq* K=8 predicts the first 8 DCT bins (a fixed mask), used to isolate the per-chunk top-K effect from the total-energy effect. (v) *Anchored Residual* of Section 4.3; its λ=0 row is the residual wrapper with the residual disabled under the stochastic-base protocol.

Spectral analysis protocol. For the offline spectral analysis (§6.1, §6.2) we apply per-dimension min-max normalization (matching training) on the train split, then compute three complementary transforms on H=16 chunks with stride 1: orthonormal DCT, real FFT (`rfft`), and Morlet wavelets (FFT-convolution CWT, w_0=6). All energies are per-chunk normalized so values are comparable across tasks of different dimensionality.

6 Results

6.1 Quantitative Evaluation

6.1.1 Q1 + Q2: A per-task K sweep finds task-dependent Sparse-DCT winners

Figure 2 shows the closed-loop success rate as a function of retained bandwidth K. The result is unambiguous: *the best K is task-dependent.*

Table 1 reports the full task–method matrix. Three patterns stand out. First, *Square is the strongest evidence for representation matters*: Raw FM is near-failure (0.24) while Sparse-DCT K=4 rescues it to 0.68. Second, *Push-T is the strongest evidence for representation hurts*: Raw FM is already strong (0.86) and only K=12 comes within striking distance (0.84). Third, the contact-heavy ManiSkill task PegInsertionSide remains weak under

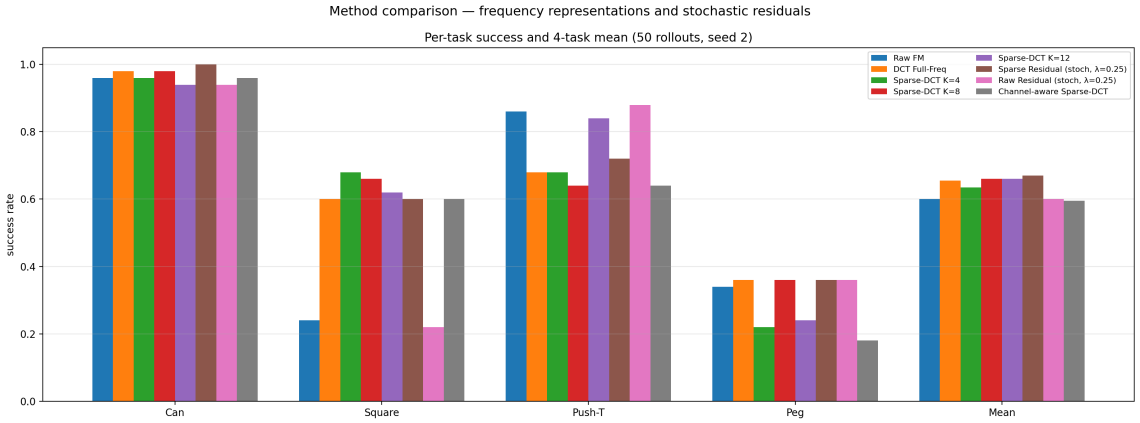

Figure 2: Mean success across the four-task suite. DCT Full-Freq improves Raw FM from 0.600 to 0.655, isolating the frequency-basis effect. The best fixed global Sparse-DCT setting reaches 0.660 (+10% relative to Raw FM), while post-hoc/oracle task-selected K reaches 0.715 (+19.2%). Within the stochastic-base residual protocol, λ=0 is best on average (0.705), and residual injection does not improve the mean.

every choice (≤ 0.36): the K knob alone does not unlock it.

Method	Can	Square	Push-T	Peg	Mean
Raw FM	0.96	0.24	**0.86**	0.34	0.600
DCT Full-Freq	0.98	0.60	0.68	0.36	0.655
Sparse-DCT K=4	0.96	0.68	0.68	0.22	0.635
Sparse-DCT K=8	0.98	0.66	0.64	0.36	0.660
Sparse-DCT K=12	0.94	0.62	0.84	0.24	0.660
Post-hoc Sparse-DCT $K^\star$	0.98	0.68	0.84	0.36	**0.715**
Stochastic base only (λ=0)	0.98	**0.70**	0.72	**0.42**	0.705
Anchored Residual (λ=0.25)	**1.00**	0.60	0.72	0.36	0.670

Table 1: Closed-loop success rate (n=50, seed 2). The $K^\star$ row is post-hoc/oracle task selection by rollout success; fixed global K=8 and K=12 both average 0.660. In the stochastic-base residual protocol, base-only execution (λ=0) beats λ=0.25 on mean success.

6.1.2 Q3: Frequency reparameterization helps, but compression is task-dependent

Table 1 separates the effect of changing basis from the additional effect of sparse truncation. DCT Full-Freq predicts all DCT coefficients and therefore tests frequency reparameterization without compression. It improves mean success over Raw FM from 0.600 to 0.655, with large gains on Square (0.24→0.60) and small gains on Can and PegInsertionSide, but a drop on Push-T (0.86→0.68). Thus the frequency basis is useful on average but not uniformly better. Sparse-DCT then adds a task-dependent compression/top-K choice: the best fixed global K reaches 0.660, only slightly above DCT Full-Freq, while post-hoc/oracle per-task $K^\star$ reaches 0.715. The gain is therefore not purely "DCT versus time domain" or purely compression; Raw FM → DCT Full-Freq measures a helpful basis-change effect, and DCT Full-Freq → Sparse-DCT measures a smaller but strongly task-dependent truncation effect.

6.1.3 Diagnostic: Energy retention is not a reliable K-selection rule

A natural hypothesis is that the best K is the smallest K for which the truncated DCT still retains most of the chunk's energy. We falsify this directly. Figure 3(a) shows the retained-energy fraction averaged over the demonstration set: every K we test retains more than 97% of total DCT energy on every task. Yet rollout success differs substantially even when total retained energy is matched.

Figure 3(b) makes the point sharper: at the same K=8, replacing the fixed Low-Freq mask with per-chunk top-K selection changes rollout success by up to 19 **points** (PegInsertionSide: Low-Freq 0.55 → Sparse 0.36) and by 14 points on Push-T (0.50 → 0.64). These two methods retain nearly identical *amounts* of energy. They differ only in *which bins* are kept. Energy is therefore the wrong axis on which to predict K; the right axis is whether the kept bins overlap with *behaviorally salient* bins.

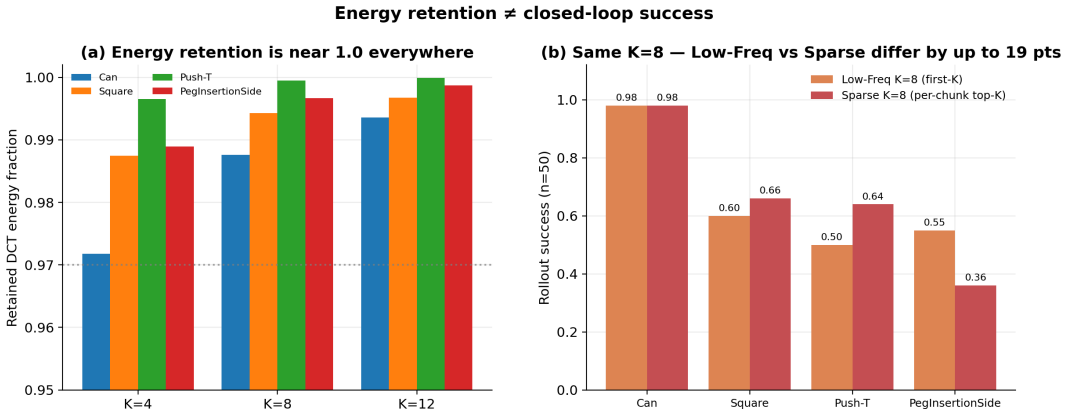

Figure 3: Diagnostic — Energy retention is decoupled from closed-loop success. (a) Top-K retained DCT energy is above 0.97 on every task for every $K \in \{4, 8, 12\}$. (b) At the *same* K=8, Low-Freq (first-K bins) versus Sparse (per-chunk top-K bins) selection diverges by up to 19 points in closed-loop success. K does not control how much energy is retained; it controls which behaviorally-salient bins are admitted.

6.1.4 Q4: Stochastic-base residual does not improve the base

The revised residual evaluation changes the Q4 answer from "partly" to essentially "no" under the stronger stochastic-base protocol. The base-only setting (λ=0) achieves the best average success (0.705). Adding the anchored residual at λ=0.25 slightly improves Can ($0.98 \rightarrow 1.00$), lowers Square ($0.70 \rightarrow 0.60$), leaves Push-T unchanged ($0.72 \rightarrow 0.72$), and lowers PegInsertionSide ($0.42 \rightarrow 0.36$), leaving the mean at 0.670. Across the full sweep, every $\lambda > 0$ is below the base-only mean.

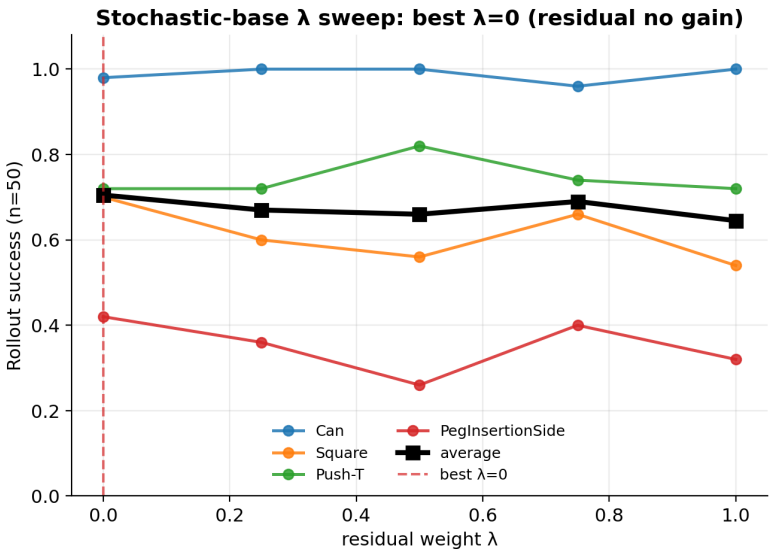

Figure 4: Q4 — Stochastic-base residual outcome. Full λ sweep across the four-task suite. The base-only policy (λ=0) has the strongest mean success. Residual weights can help individual tasks at isolated values, but no tested residual weight improves average performance over the base.

Direct answer to Q4. The learned frequency residual does not reliably recover useful discarded bins. It is useful diagnostically because it isolates the bandwidth-correction question, but the stronger stochastic-base sweep shows that one global residual weight and one global gate are too coarse for task-, channel-, and phase-local bandwidth needs.

6.1.5 Expert action spectra are strongly low-frequency on every task

A direct offline analysis of the expert demonstrations confirms the project's core premise. Three complementary transforms — orthonormal DCT, FFT, and Morlet wavelets — all agree that demonstration action chunks are dominated by their lowest-frequency component on every task (Figure 5). The lowest DCT coefficient alone carries 80−93% of total energy, and the first K=4 coefficients carry $\geq 97\%$ on every task (Table 2).

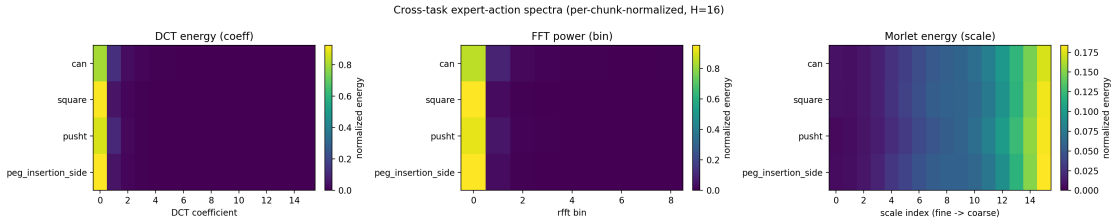

Figure 5: Cross-task expert-action spectra (per-chunk normalized, H=16). DCT, FFT, and Morlet decompositions all concentrate energy at the lowest frequencies / coarsest scales on every task. Energy locks onto DCT coefficient 0 / FFT bin 0.

Task	DC (k=0)	cumE@K=4	cumE@K=8	Recon MSE@K=4	FFT bin 0
Can	0.802	0.972	0.988	0.0071	0.860
Square	0.920	0.987	0.994	0.0043	0.954
Push-T	0.869	**0.997**	1.000	0.0003	0.920
PegInsertionSide	0.925	0.989	0.997	0.0028	0.953

Table 2: Per-task expert-action spectral concentration. Every task retains $\geq 97\%$ of DCT energy at K=4; Push-T is the most compressible by both DCT and FFT measures.

The phase-conditioned spectrum explains why the global energy table is not sufficient for choosing K. Most H=16 chunks are smooth non-transition windows, so the averaged spectrum is dominated by the DC and first few low-frequency bins. When we split Robomimic chunks by whether the normalized gripper command changes by more than 1.0 within the chunk, the residual high-frequency mass becomes concentrated in a small but behaviorally important subset. For Can, transition chunks have 0.079 aggregate DCT energy above bin 4 versus 0.008 for non-transition chunks; for Square the same comparison is 0.040 versus 0.006. The gripper channel is the clearest example: its high-frequency ratio is about 0.115 during transitions and rounds to 0.000 outside transitions, while rotation stays near 0.001 in both phases (Figure 6). Thus the relevant frequency structure is *phase-local and channel-local*: a single global K can smooth the common approach motion, but it cannot selectively allocate extra bins to gripper/translation exactly at contact or open/close events.

6.1.6 Demonstration spectrum partially predicts closed-loop best K

A natural hypothesis is that the closed-loop best K matches the smallest K at which top-K reconstruction of the expert chunks saturates — i.e., the spectrum's effective bandwidth predicts the policy's optimal bandwidth. The hypothesis holds for three of four tasks: Can, Square, and Peg fit within ± 2 bins of their reconstruction knees. **Push-T is a clean counterexample**: it is the *most* compressible task in our suite (99.7% energy at K=4) yet its closed-loop best K is the *largest* (K=12). This implies that K controls more than reconstruction fidelity on the demo distribution — policy expressivity in the truncated basis and out-of-distribution recovery during rollout also matter, and a pure demo-side metric cannot fully predict K alone.

6.1.7 Residual λ sweep

Because λ is an execution-time scalar, the trained residual can be evaluated at multiple residual weights without retraining. The full stochastic-base sweep gives:

The sweep shows task-specific behavior rather than a useful global residual setting: Can is saturated, Push-T briefly improves at λ=0.5, Square and Peg prefer no residual, and the average best is λ=0. This supports treating the residual as a diagnostic rather than as the final method.

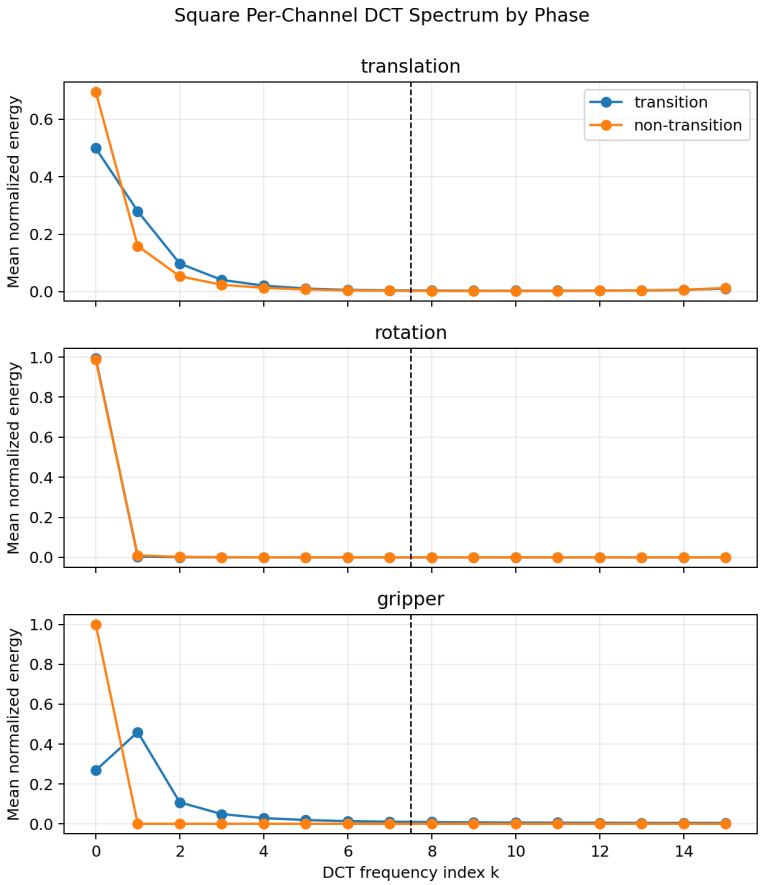

Figure 6: Phase-conditioned DCT spectrum for Square, split by channel group. Gripper-transition chunks expose the small high-frequency tail that is hidden in the global spectrum. Rotation channels remain nearly pure low-frequency in both phases, while gripper and translation allocate visibly more mass to higher DCT bins during open/close and contact-transition windows.

Task	λ=0	0.25	0.5	0.75	1.0
Can	0.98	**1.00**	**1.00**	0.96	**1.00**
Square	**0.70**	0.60	0.56	0.66	0.54
Push-T	0.72	0.72	**0.82**	0.74	0.72
PegInsertionSide	**0.42**	0.36	0.26	0.40	0.32
Average	**0.705**	0.670	0.660	0.690	0.645

Table 3: Full stochastic-base λ sweep (n=50, seed 2). The best average is base-only execution (λ=0); every residual setting is lower on average.

6.1.8 Multi-seed sensitivity

The primary reported numbers in this section are single-seed (seed 2, n=50). On the Robomimic subset, an earlier sweep across seeds $0, 1, 2$ yields multi-seed means (Can: Raw 0.96, K=8 Sparse 0.95, Full DCT 0.98; Square: Raw 0.45, K=4 Sparse 0.63, K=12 Sparse 0.54). The qualitative Robomimic ordering — Square strongly prefers low K and Can is flat-to-improved — is preserved across seeds. The *magnitudes* are not: Square's K=4 ranges from 0.32 to 0.90 across the three seeds. Push-T and PegInsertionSide remain single-seed headline results in this report. Three seeds are not enough to claim per-task optimal K at the granularity of single rollout points; we mark this as a limitation (§7).

6.2 Qualitative Analysis

We complement the quantitative results with three qualitative lenses: how policies actually fail in rollout, the per-channel structure of expert action spectra (and what it implies for the residual gate), and the gate's learned behavior.

6.2.1 Failure-mode patterns

Across the brittle tasks (Square, PegInsertionSide, Push-T), the overwhelming failure mode is *timeout*, not immediate crash. Policies typically run through approach and contact-like behavior but fail to complete the precise final geometric condition. On Square, Raw FM has 38/50 timeout failures whereas Sparse-DCT K=4 has 16/50, with average successful-episode steps dropping from 350.5 to 246.6. Two qualitative observations follow.

First, the additional bandwidth that residual injection might help with is needed in the *precision/contact substage*, not in early reach. This argues against a single global gating scalar λ applied uniformly across the chunk: the bandwidth-recovery problem is phase-localized, but our gate $g(o_t)$ has no phase input. *Second*, on PegInsertionSide, residual injection increases failures/timeouts relative to the λ=0 stochastic base ($0.42 \rightarrow 0.36$ success). On Push-T, λ=0.25 leaves success unchanged relative to λ=0 ($0.72 \rightarrow 0.72$), although larger λ values show task-specific tradeoffs. This pattern is consistent with a residual that is sometimes useful but too globally timed for late-stage precision phases — matching the "conservative but mistimed" gate diagnostic of §6.2.3.

6.2.2 Per-channel structure: rotation has a near-zero high-frequency tail

Splitting the DCT by channel group exposes a sharp asymmetry (Figure 6). On the Robomimic tasks, rotation channels carry essentially no high-frequency energy (high-freq ratio ≈ 0.001 above K=4), while translation and gripper channels carry what little high-frequency content exists. The phase-conditioned analysis above sharpens this: the extra bandwidth is concentrated during gripper-transition windows rather than spread uniformly across the trajectory. This asymmetry is qualitatively important for the residual question. A single global K either *wastes capacity* on rotation (where any $K > 1$ adds almost no useful content) or *under-allocates* to gripper transitions (where K=4 throws away most of the transition-window bandwidth). A residual that is gated uniformly across channels is biased toward arm-channel corrections it cannot helpfully use, while leaving the gripper channel under-served exactly when its short transition events demand more bins.

6.2.3 Learned gate behavior

Across all four tasks the learned soft frequency gate $g(o_t)$ converges to small values ($\bar{g} \approx 0.055$–0.086), giving an *effective* retained bandwidth from the residual of less than 1.4 bins per channel on every task (and below 1 bin on PegInsertionSide). Qualitatively this means the residual is *conservative by construction* — the gate's ℓ_1 prior and small initialization pull it toward inactivity, and the FM loss provides no strong signal to push it back up because the training objective is dominated by reconstructing the gated DCT residual rather than the time-domain action. The gate's small-but-non-zero values appear unrelated to any phase signal: it activates roughly uniformly across the chunk rather than concentrating at the precision/contact substage where §6.2.1 indicates the bandwidth is actually needed. This explains the apparent paradox of Q4: the residual is simultaneously *too small to help* on the tasks whose base is already strong (Can) and *badly timed* on the tasks where any non-zero correction adds jitter (Push-T, Peg).

7 Discussion

Takeaway 1: Representation selection beats representation invention. The best single number we report — post-hoc/oracle task-selected mean success 0.715 versus Raw FM 0.600 — comes from selecting K per task, not from any new model component. DCT Full-Freq already improves the mean to 0.655, so the frequency-basis change helps on average; the best fixed Sparse-DCT setting is only slightly higher at 0.660, while the oracle $K^\star$ row should be read as a diagnostic selected by rollout success. Within the stochastic-base residual protocol, λ=0 reaches 0.705, while all residual weights are lower. In the small-budget regime characteristic of class projects, controlled representation sweeps are a more reliable source of gain than new gating components.

Takeaway 2: Energy retention is the wrong axis at fixed chunking. A small, behaviorally critical fraction of DCT energy (gripper transitions, contact onsets, late recovery corrections) is what drives closed-loop

success, and it is invisible to the energy axis. This is a warning against tuning chunk truncation by reconstruction quality alone, even on demonstration data with high SNR.

Takeaway 3: A globally-gated residual is a diagnostic, not a unified method. The anchored residual does not improve average success over its stochastic base. The learned gate is small but apparently mistimed; absent phase or contact conditioning, the residual cannot be applied only where the base is wrong. We retain it as a useful diagnostic — it isolates the bandwidth-recovery question — and the full stochastic λ sweep (§6.1.7) shows no global residual weight beats base-only execution.

Why per-channel is the next lever. The qualitative per-channel analysis (§6.2.2) points to a concrete next step: assign different K (or different residual-gate budgets) to the rotation, translation, and gripper channels rather than choosing a single global K. The asymmetry is large enough — rotation's high-frequency ratio is two orders of magnitude below gripper's — that the global choice almost certainly either wastes capacity or under-allocates to behaviorally critical bins.

Limitations. (i) Single-seed headline results on all four tasks; multi-seed sensitivity is available only on the Robomimic subset, and the magnitude variance is substantial (Square's K=4 ranges 0.32–0.90). (ii) No real-robot evaluation: all results are in simulation. (iii) The Peg ceiling (≤ 0.36 across every method tested) is not solved by any frequency intervention; the bottleneck is more likely backbone capacity or observation modality than action representation. (iv) The anchored residual was trained at a single λ=0.25. The full stochastic λ sweep (§6.1.7) shows no global residual weight improves the average. (v) The per-channel and per-phase K ablations motivated by §6.2.2 are not yet run.

8 Conclusion

We investigated frequency-domain action representations for Flow Matching robot policies on four manipulation benchmarks. Three findings emerge.

First, frequency reparameterization helps on average: DCT Full-Freq raises mean closed-loop success from 0.600 (Raw FM) to 0.655. Sparse-DCT compression adds a task-dependent knob: the best fixed global bandwidth reaches 0.660 (+10% relative), while post-hoc/oracle per-task bandwidth selection reaches 0.715 (+19.2%); no universal K wins across tasks. Second, energy retention is the wrong axis for choosing K: at matched K and similar retained energy, Low-Freq and per-chunk Sparse selection can differ by up to 19 points, showing that behaviorally salient bin identity — not total retained energy — is the key axis. Third, under the stochastic-base protocol, a conservative DCT residual at λ=0.25 slightly improves Can, hurts Square and PegInsertionSide, leaves Push-T unchanged, and lowers mean success from 0.705 to 0.670; the globally-gated correction is too coarse for the contact-precision tasks.

A spectral analysis of expert demonstrations confirms strong low-frequency concentration on every task ($\geq 97\%$ energy in K=4) and reveals a per-channel asymmetry — rotation has a near-zero high-frequency tail, while gripper and translation carry the small high-frequency content. This directly motivates per-channel rather than global bandwidth selection as the most promising next step. Multi-seed validation, phase- or contact-conditioned residual gating, and real-robot evaluation remain open.

The broader takeaway is that representation selection is a more reliable source of gain than residual gating in this small-budget regime; controlled per-task sweeps are a defensible default for action-chunk policies whose backbone is fixed, and the per-channel asymmetry of the expert spectrum is the most actionable signal for the next round of method design.

9 Team Contributions

Alan Zhao and Juhyun Jung contributed equally to the project. Alan led the Flow Matching / DCT policy implementation, residual-wrapper experiments, and rollout evaluation infrastructure. Juhyun led the spectral and phase-conditioned analyses, figure/table generation, and report writing. Both authors jointly designed the research questions, checked numerical claims against evaluation artifacts, and edited the final report. This division remained broadly consistent with the original proposal.

10 AI Tools Disclosure

Per the course AI-tools policy: an AI coding assistant (Claude, via Claude Code) was used for engineering and writing support — scaffolding the new analysis code (FFT/Morlet transforms, the unified spectral runner, heatmap plotting, and the λ-sweep driver), orchestrating training/evaluation runs, aggregating results, and drafting this

report. The research design, method, hyperparameters, and interpretation were directed by the author; all numbers in this report come from actual training and 50-rollout evaluation runs (artifacts under `outputs/analysis/`, `outputs/train/`, and `outputs/eval/`), not from the assistant. This final report was drafted by the authors and then edited with AI assistance for tightening prose, formatting LaTeX tables and figure environments.

References

[1] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. *Robotics: Science and Systems (RSS)*, 2023. Extended in *International Journal of Robotics Research*, 2024.

[2] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware (ACT / ALOHA). *Robotics: Science and Systems (RSS)*, 2023.

[3] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow Matching for Generative Modeling. *International Conference on Learning Representations (ICLR)*, 2023.

[4] X. Liu, C. Gong, and Q. Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. *International Conference on Learning Representations (ICLR)*, 2023.

[5] K. Pertsch *et al.* FAST: Efficient Robot Action Tokenization via Frequency-Space Compression for VLA Models. *arXiv preprint arXiv:2501.09747*, 2025.

[6] S. Belkhale, Y. Cui, and D. Sadigh. HYDRA: Hybrid Robot Actions for Imitation Learning. *Conference on Robot Learning (CoRL)*, 2023.

[7] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine. Residual Reinforcement Learning for Robot Control. *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[8] T. Silver, K. Allen, J. Tenenbaum, and L. Kaelbling. Residual Policy Learning. *arXiv preprint arXiv:1812.06298*, 2018.

[9] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What Matters in Learning from Offline Human Demonstrations for Robot Manipulation (Robomimic). *Conference on Robot Learning (CoRL)*, 2021.

[10] J. Gu *et al.* ManiSkill2: A Unified Benchmark for Generalizable Manipulation Skills. *International Conference on Learning Representations (ICLR)*, 2023.

[11] J. Makhoul. A Fast Cosine Transform in One and Two Dimensions. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(1):27–34, 1980.