PANDORA: Diffusion Policy Learning for Dexterous Robotic Piano Playing with a Train-only LLM Expressiveness Reward

Yanjia Huang[1] **Renjie Li**[1] **Zhengzhong Tu**[1*]
[1]Texas A&M University, College Station, TX, USA
{yanjia_0812,renjie,tzz}@tamu.edu

Abstract

Robotic piano performance requires both accurate key strikes and *expressiveness*—human-like variation in timing, dynamics, articulation, and left/right-hand coordination. We present **PANDORA**, a diffusion-based policy for dexterous control trained with a *train-only* language-model expressiveness reward. From each rollout we compute *symbolic* descriptors (tempo curve via IOI-CV/nPVI, velocity histogram, articulation ratio), serialize them as compact JSON, and query an off-the-shelf LLM with a 0–4 rubric (deterministic, cached). The normalized score shapes the total reward, while inference never queries the oracle. A residual inverse-kinematics controller enforces joint/velocity limits. On *RoboPianist*, PANDORA improves note accuracy and perceived phrasing compared to PianoMime variants. [2]

1 Introduction

Robotic musicianship demands more than accurate key presses. Existing methods trained by imitation or reinforcement learning often sound mechanical: quantized timing, flat dynamics, and lack of expressiveness [2, 6, 16]. Two observations motivate our approach. **(i)** Diffusion policies suit high-rate continuous control: denoising refines actions without compounding auto-regressive error. **(ii)** Human notions of "musicality" are hard to encode by hand yet can be judged from compact statistics by language models [9, 20]. We propose **PANDORA**, a diffusion-guided pianist trained with a **language-model expressiveness reward**. During *training only*, phrase-wise symbolic descriptors—IOI-CV/nPVI (tempo curve), velocity histogram, articulation ratio—are serialized to JSON and scored by an off-the-shelf LLM on a 0–4 rubric (deterministic decoding; cached outputs) [3, 7, 12]. The normalized score shapes the reward; inference never queries the oracle. By avoiding *raw audio*, the reward is insensitive to simulator timbre and is easy to reproduce. In summary, PANDORA seeks to bridge a longstanding gap in dexterous manipulation: the disparity between precise mechanical control and the nuanced interpretive qualities required for artistic expression. Through an innovative synergy between diffusion-based policy learning and LLM-driven semantic rewards, our PANDORA framework not only achieves precise key-presses but also imbues robotic piano performances with genuine musical expressiveness. Our key contributions are summarized as follows:

- We introduce a diffusion policy learning approach featuring a conditional U-Net with FiLM-based global conditioning, generating robust, high-dimensional action trajectories reflective of human expressivity.

*Corresponding author.
[2]Projects can be found: https://taco-group.github.io/PANDORA/

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI for Music.

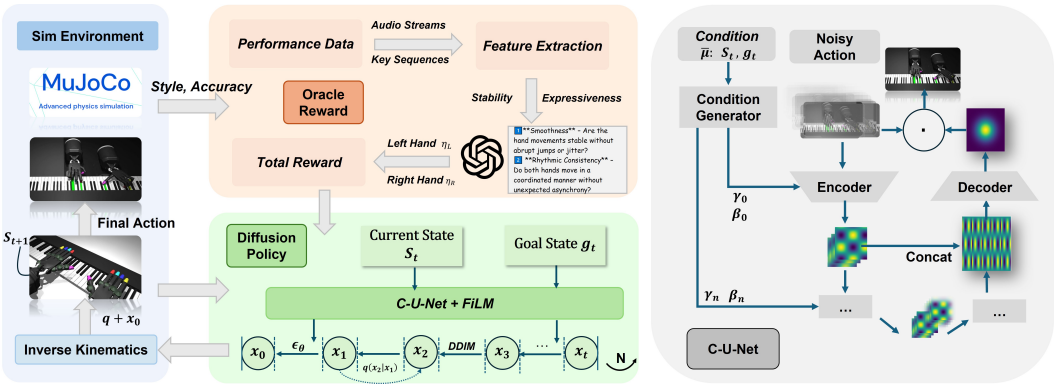


Figure 1: State and goal conditioned diffusion ([11], [13]) produces x_0, executed via residual IK in MuJoCo [19]; an LLM oracle scores symbolic expressiveness (train-only).

- We introduce a novel composite reward function that leverages a large language model oracle to provide semantic feedback, thereby enriching the learning process with qualitative artistic insights beyond traditional numeric metrics.
- We incorporate a residual inverse-kinematics refinement policy that enhances fine-grained finger-level precision, significantly improving the robotic system's ability to execute complex and expressive piano techniques.
- Through extensive experimentation and rigorous ablation studies in the ROBOPIANIST environment, we empirically demonstrate that PANDORA achieves superior performance in both technical precision and musical expressiveness compared to existing state-of-the-art methods.

2 Method

2.1 State, Goal and Action Parameterization

At time t, the agent observes $s_t{=}(q_t, \dot{q}_t, \text{key states})$ and a short-horizon goal g_t (a queue of target key events within the next 0.3–0.5 s). The policy outputs an *action chunk* $x_0 \in \mathbb{R}^{H\times d}$ (duration $H/100$ s at 100 Hz) which is executed by a residual inverse-kinematics (IK) controller to update joints $q_{t+1} = q_t + \Delta q$ while enforcing joint/velocity limits (penalized in r_{task}). We keep all simulator and sensing components identical to the baseline to isolate our contributions.

2.2 Conditioned Diffusion Policy

Architecture. A condition encoder maps $\tilde{u}_t = [s_t, g_t]$ to FiLM [11] parameters (γ, β) that modulate each block of a conditional U-Net ϵ_θ. Given a noise level $t \in \{1, \dots, N\}$, the network predicts $\hat{\epsilon} = \epsilon_\theta(x_t, t \mid \tilde{u}_t)$ and we perform a DDIM [4, 5, 15] step

$$x_{t-1} = \text{DDIM}(x_t, \hat{\epsilon}, t;\, \eta{=}0),$$

yielding x_0 after N steps (we use $N{=}20$ by default) [13]. This denoising view stabilizes high-rate control and affords sub-beat timing without compounding auto-regressive error.

PianoMime anticipated DDIM as an engineering improvement; we *integrate* diffusion with (i) explicit FiLM conditioning on *state+goal*, and (ii) residual IK [18] (below), which together make diffusion feasible for dexterous piano where timing jitter and kinematic feasibility are tightly coupled.

2.3 Residual IK with Physical Constraints

Given x_0 (end-effector deltas), IK solves a small QP

$$\min_{\Delta q} \; \|J(q_t)\Delta q - x_0\|_2^2 \;+\; w_{\text{jl}}\phi_{\text{jl}}(q_t{+}\Delta q) \;+\; w_{\text{vel}}\|\Delta q\|_2^2,$$

where ϕ_{jl} penalizes joint-limit violations [10]; max-velocity is enforced softly. The terms $w_{\text{jl}}, w_{\text{vel}}$ are the same across all methods and their penalties are part of r_{task} [1, 8, 14, 17]. *Design choice.* We use the residual form by default (more stable than direct joint prediction); we treat it as an engineering choice, not a novelty, but we make its *interaction* with diffusion explicit and reproducible.

2.4 Train-only LLM Expressiveness Oracle

Descriptor design (no raw audio). From each rollout we compute *phrase-wise* (window 3.0 s, hop 1.5 s) symbolic statistics, optionally per hand: (i) **tempo curve**: IOI-CV and nPVI; (ii) **velocity**: mean/std and 8-bin histogram (Gaussian smoothing σ=1 bin) plus temporal slope; (iii) **articulation ratio** d_k/IOI_k with legato/staccato fractions. Only these descriptors are passed to the oracle; *no audio* is used, avoiding simulator timbre confounds.

Prompting & rubric. An off-the-shelf LLM (GPT-4o) is prompted as a pedagogy rater with *K=4* few-shot exemplars and a **0–4** rubric (poor→excellent) that explicitly references (a) rhythmic stability, (b) dynamic shaping, (c) LH/RH coordination. Decoding is deterministic (`temperature`=0.0, `top_p`=1.0, `max_tokens`=64) and returns strict JSON: {`score` $\in \{0..4\}$, `rationale`}. Queries are cached by SHA-256 over the JSON + rubric version to ensure bit-wise repeatability; the oracle is **never** queried at inference.

Score fusion and reward shaping. Per-hand scores $s_{LH}, s_{RH} \in \{0, \ldots, 4\}$ merge to

$$s = \tfrac{1}{2}(s_{LH} + s_{RH}), \qquad \tilde{s} = s/4 \in [0, 1], \quad r_{\text{total}} = r_{\text{task}} + \lambda_{\text{expr}}\tilde{s},$$

with λ_{expr}=0.10 by default. This injects human-aligned phrasing/dynamics/coordination while preserving accuracy.

Compact schema (for reproducibility). We serialize each phrase as:

```
{ "ioi_cv": 0.11, "npvi": 41.7,
  "vel_hist": [0.02,0.05,0.11,0.22,0.26,0.20,0.10,0.04],
  "artic_ratio": 0.63, "hand": "LH" }
```

and prompt with 4 rubric bullets (0..4) requesting a single integer `score` and a one-line `rationale`.

2.5 Training objective and defaults

We train the conditional diffusion model with the standard denoising objective and optimize expected r_{total}; unless stated, we use DDIM-20 (η=0), 100 Hz control with 0.10 s chunks, 3.0/1.5 s phrase window/hop, 8-bin velocity histogram, K=4 exemplars, and λ_{expr}=0.10.

3 Experiments & Qualitative

Setup. We evaluate on *RoboPianist* with the same splits as prior work. The agent observes simulated hand/key states and a short-horizon goal. Actions are executed by residual IK with joint/velocity penalties included in r_{task}. Training queries the oracle phrase-wise; inference never queries the oracle.

Baselines. **PianoMime (two-stage diff)** and a residual variant (engineering improvement). Our full method **PANDORA** adds the LLM reward to diffusion+residual control.

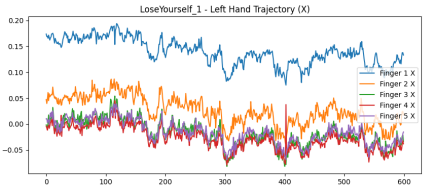
Metric. We report note-level **F1**. Predicted note onsets/offsets (from rendered MIDI) are matched to the reference score using a $\pm$40 ms onset tolerance and a 20% relative tolerance on duration; precision/recall are computed per piece and then macro-averaged over three seeds. We qualitatively discuss tempo variability (nPVI) trends on the project page. [3]

Finger Trajectory Comparison To evaluate the effectiveness of our approach in controlling hand movements, we analyze the X-axis trajectories of each finger during piano performance. Figure 2illustrates the trajectory comparisons between different methods.

[3] Per-song results are provided in the supplement/webpage. Rhythm variability (nPVI) trends are discussed qualitatively with audio demos.

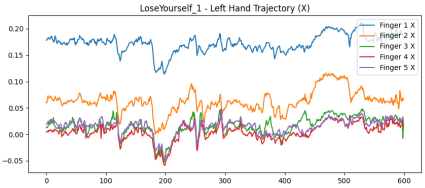
Table 1: **Quantitative results of each song in our collected test dataset.**

Song Name	Two-stage Diff (PianoMime)			Two-stage Diff-res (PianoMime)			PANDORA (Ours)		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Forester	**0.81**	0.70	0.68	0.79	0.71	0.67	**0.81**	**0.75**	**0.78**
Wednesday	0.66	0.57	0.58	**0.67**	0.54	0.55	**0.79**	**0.62**	**0.70**
Alone	0.80	0.62	0.66	**0.83**	0.65	0.67	0.80	**0.66**	**0.72**
Only We Know	0.63	0.53	0.58	0.67	0.57	0.59	**0.73**	**0.67**	**0.70**
Eyes Closed	0.60	0.52	0.53	**0.61**	0.45	0.50	0.59	**0.53**	**0.56**
Pedro	0.70	0.58	**0.60**	0.67	0.56	0.47	**0.78**	**0.64**	**0.70**
Ohne Dich	0.73	0.55	0.58	0.75	**0.56**	**0.62**	**0.79**	0.55	0.65
Paradise	0.66	0.42	0.43	0.68	0.45	**0.47**	**0.77**	**0.48**	**0.59**
Hope	0.74	0.55	0.57	0.76	0.58	0.62	**0.81**	**0.62**	**0.70**
No Time To Die	0.77	0.53	0.55	0.79	**0.57**	**0.60**	**0.88**	**0.57**	**0.69**
The Spectre	0.64	**0.52**	**0.54**	**0.67**	0.50	0.52	0.63	0.49	0.53
Numb	0.55	0.44	0.45	0.57	0.47	0.48	**0.67**	**0.49**	**0.57**
Cold Play	0.63	0.33	0.64	/	/	/	**0.67**	**0.50**	**0.74**
Mean	0.68	0.54	0.57	0.70	0.56	0.58	**0.78**	**0.60**	**0.68**



(a) Left Hand (Ours)

(b) Right Hand (Ours)



(c) Left Hand (PianoMime)

(d) Right Hand (PianoMime)

Figure 2: **Finger-trace comparison.** Ours vs. PianoMime for left/right hands (top vs. bottom).

4 Discussion, Limitations, and Conclusion

Discussion. Conditioning diffusion on state+goal and executing via residual IK yields stable sub-beat timing while respecting joint/velocity limits; a *train-only*, deterministic LLM oracle that scores *symbolic* descriptors (IOI-CV, nPVI, velocity histogram, articulation) supplies a reproducible expressiveness signal insensitive to simulator timbre. Empirically we observe higher F1 and increased nPVI, with clearer LH–RH alternation in overlays.

Limitations. The 0–4 rubric and feature set are subjective and coarse (e.g., no pedaling/voice-leading); λ_{expr} and window/hop trade-offs are only partially explored; very fast leaps remain failure modes and sim-to-real is untested; no formal listener study is included.

Conclusion. PANDORA couples conditioned diffusion and residual IK with a deterministic, train-only symbolic oracle, improving accuracy and phrasing while remaining reproducible via released schema, rubric, and cache keys.

References

[1] Ren Allen, Z., Lidard Justin, Ankile Lars, L., Simeonov Anthony, Agrawal Pulkit, Majumdar Anirudha, Burchfiel Benjamin, Dai Hongkai, and Simchowitz Max. Diffusion policy policy optimization. *arXiv preprint arXiv:2409.00588*, 2024. URL https://www.arxiv.org/abs/2409.00588.

[2] Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models, 2023. URL https://arxiv.org/abs/2310.10639.

[3] Morales-Brotons Daniel and and Hadrien Hendrikx Thijs, Vogels. Exponential moving average of weights in deep learning: Dynamics and benefits. *arXiv preprint arXiv:2411.18704*, 2024. URL https://www.arxiv.org/abs/2411.18704.

[4] Zhang Edwin, Lu Yujie, Wang William, and Zhang Amy. Language control diffusion: Efficiently scaling through space, time, and tasks. *arXiv preprint arXiv:2210.15629*, 2022. URL https://www.arxiv.org/abs/2210.15629.

[5] Saha Kallol, Mandadi Vishal, Reddy Jayaram, Srikanth Ajit, Agarwal Aditya, Sen Bipasha, and and Madhava Krishna Arun, Singh. Edmp: Ensemble-of-costs-guided diffusion for motion planning. *arXiv preprint arXiv:2309.11414*, 2023. URL https://www.arxiv.org/abs/2309.11414.

[6] Sulabh Kumra, Shirin Josh, and Ferat Sahin. Learning robotic manipulation tasks via task progress based gaussian reward and loss adjusted exploration, 2021. URL https://arxiv.org/abs/2103.01434.

[7] Ahn Kwangjun and Cutkosky Ashok. Adam with model exponential moving average is effective for nonconvex optimization. *arXiv preprint arXiv:2405.18199*, 2024. URL https://www.arxiv.org/abs/2405.18199.

[8] Reuss Moritz, Yağmurlu Ömer, Erdinç, Wenzel Fabian, and Lioutikov Rudolf. Multimodal diffusion transformer: Learning versatile behavior from multimodal goals. *arXiv preprint arXiv:2407.05996*, 2024. URL https://www.arxiv.org/abs/2407.05996.

[9] Shivansh Patel, Xinchen Yin, Wenlong Huang, Shubham Garg, Hooshang Nayyeri, Li Fei-Fei, Svetlana Lazebnik, and Yunzhu Li. A real-to-sim-to-real approach to robotic manipulation with vlm-generated iterative keypoint rewards, 2025. URL https://arxiv.org/abs/2502.08643.

[10] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions On Graphics (TOG)*, 37(4):1–14, 2018.

[11] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer, 2017. URL https://arxiv.org/abs/1709.07871.

[12] Li Siyuan, Liu Zicheng, Tian Juanxi, Wang Ge, Wang Zedong, Jin Weiyang, Wu Di, Tan Cheng, Lin Tao, Liu Yang, Sun Baigui, and Z. Li and, Stan. Switch ema: A free lunch for better flatness and sharpness. *arXiv preprint arXiv:2402.09240*, 2024. URL https://www.arxiv.org/abs/2402.09240.

[13] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

[14] Lee Sung-Wook and Kuo Yen-Ling. Diff-dagger: Uncertainty estimation with diffusion policy for robotic manipulation. *arXiv preprint arXiv:2410.14868*, 2024. URL https://www.arxiv.org/abs/2410.14868.

[15] Huang Tao, Jiang Guangqi, Ze Yanjie, and Xu Huazhe. Diffusion reward: Learning rewards via conditional video diffusion. *arXiv preprint arXiv:2312.14134*, 2023. URL https://www.arxiv.org/abs/2312.14134.

[16] Jianren Wang, Sudeep Dasari, Mohan Kumar Srirama, Shubham Tulsiani, and Abhinav Gupta. Manipulate by seeing: Creating manipulation controllers from pre-trained representations, 2023. URL https://arxiv.org/abs/2303.08135.

[17] Cao Yuhong, Lew Jeric, Liang Jingsong, Cheng Jin, and Sartoretti Guillaume. Dare: Diffusion policy for autonomous robot exploration. *arXiv preprint arXiv:2410.16687*, 2024. URL https://www.arxiv.org/abs/2410.16687.

[18] Kevin Zakka, Philipp Wu, Laura Smith, Nimrod Gileadi, Taylor Howell, Xue Bin Peng, Sumeet Singh, Yuval Tassa, Pete Florence, Andy Zeng, et al. Robopianist: Dexterous piano playing with deep reinforcement learning. *arXiv preprint arXiv:2304.04150*, 2023.

[19] Kevin Zakka, Baruch Tabanpour, Qiayuan Liao, Mustafa Haiderbhai, Samuel Holt, Jing Yuan Luo, Arthur Allshire, Erik Frey, Koushil Sreenath, Lueder A. Kahrs, Carmelo Sferrazza, Yuval Tassa, and Pieter Abbeel. Mujoco playground, 2025. URL https://arxiv.org/abs/2502.08844.

[20] Yanjie Ze, Ge Yan, Yueh-Hua Wu, Annabella Macaluso, Yuying Ge, Jianglong Ye, Nicklas Hansen, Li Erran Li, and Xiaolong Wang. Gnfactor: Multi-task real robot learning with generalizable neural feature fields, 2024. URL https://arxiv.org/abs/2308.16891.