Natural Dexterous Piano Playing at Scale With Video Hand Priors

Jeff Tan, Yuanhao Wang, Haoyang He
Robotics Institute
Carnegie Mellon University
{jefftan, yuanhao4, hhe2}@andrew.cmu.edu

Abstract: Building robotic hands with human-like dexterity is one of the most important open problems in robotics. Despite tremendous research, recent methods are limited to a narrow set of dexterous tasks such as object grasping and in-hand cube manipulation. Although more challenging tasks such as robotic piano playing have been recently demonstrated, existing RL approaches are unable to play arbitrary pieces zero-shot, and are limited to playing a specific 30-second piece given dense expert fingering labels as input. To improve the scalability of this system and avoid the need for expert labeling, we introduce a method to learn piano playing directly from widely-available YouTube videos, by generating automated fingering labels with state-of-the-art hand pose estimation and music note transcription. Our method is able to learn a challenging 14-minute long piano piece by copying the fingering from human videos, enabling large-scale training data generation for zero-shot piano playing at scale.

1 Introduction

Building robotic hands with human-like dexterity is one of the most important open problems in robotics. Despite tremendous research, recent methods are limited to a narrow set of dexterous capabilities such as object grasping, often with few degrees of freedom and a single goal state. For more challenging tasks such as robotic piano playing, where there are many degrees of freedom and multiple simultaneous goals, robotic approaches remain far below the level of human capabilities.

To make progress on such a difficult task, prior work on robotic piano playing (e.g. RoboPianist [1]) has typically introduced dense rewards to provide an intermediate supervision signal, such as a reward for hitting the correct piano key with a specific finger. Dense rewards make it easier to explore such a complex action space, but this requirement for dense expert fingering labels severely limits the scalability of RoboPianist. How can we bestow robots with the ability to learn from vast collections of Internet-scale data, without the requirement for expert data labels?

Problem Statement: In this project, we aim to explore whether robots can learn natural, human-like piano playing by imitating human hand motions from video demonstrations. As expert fingering labels are not readily available for real piano pieces, we develop a system that uses state-of-the-art hand reconstruction and music note transcription to recover fingering patterns automatically. We show that our method is able to play substantially longer piano pieces than prior art by watching human videos.

2 Related Work

RoboPianist [1] is the most relevant work to our idea, as we are not aware of any prior work that tries to learn human piano playing policies by watching videos. Given an input piano piece formatted as a MIDI file, RoboPianist instructs two Shadow Hands to play the piece within a simulated piano playing environment built in MuJoCo [2]. RoboPianist formulates the piano playing problem as a

Course Project, 16-831 Introduction to Robot Learning, Spring 2024, CMU

finite-horizon Markov decision process, where the reward has two parts: (1) a sparse reward for hitting the correct key at the correct time, and (2) a dense reward that encourages the fingers to be spatially close to the keys they need to press (as prescribed by the fingering labels). However, this requires expert fingering labels per note, which is not scalable as it is rarely available for the entire piano piece. For example, RoboPianist used the PIG dataset [3], which only includes 150 1-minute music pieces. We hope to mostly use the same RL formulation aside from a few key differences. In particular, we hope to augment the dense finger-to-key distance reward by imitating human hand poses from videos.

3 Method

The goal of our method is to improve the scalability of existing RL methods for robotic piano playing [1], by removing the requirement for expert fingering labels. Towards this goal, we propose a pipeline that combines 3D human hand reconstruction and music note transcription to generate automated fingering labels for an arbitrary YouTube piano playing video. Fig. 1 provides a summary of our architecture.

Intuitively, piano playing is a complicated task that involves difficult decisions about which fingering patterns to use. Providing fingering guidance, whether through manual expert labeling or automated methods, should improve the quality of piano playing methods. As this expert labeling process is incredibly tedious, automated fingering labelers are much more scalable and allow finger labels to be generated for much longer pieces. Hand reconstruction can also provide a signal to learn the more subjective and natural features of piano playing, such as expressive wrist and forearm motions.

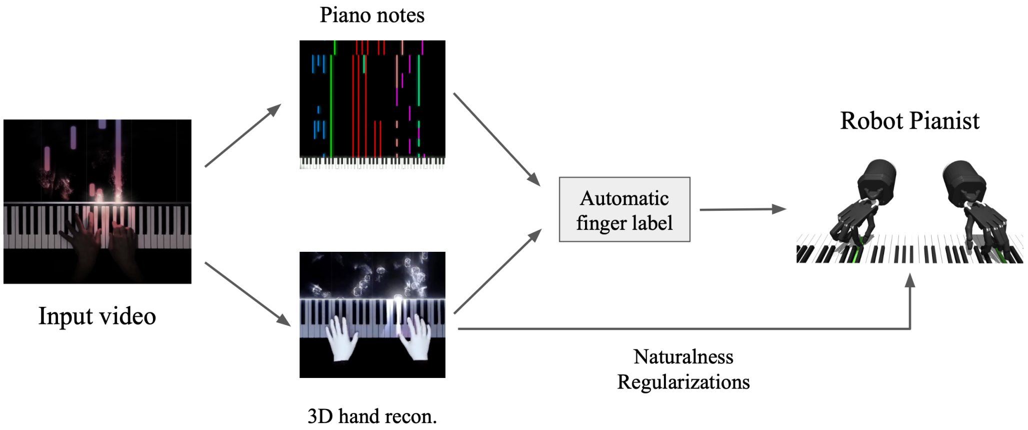
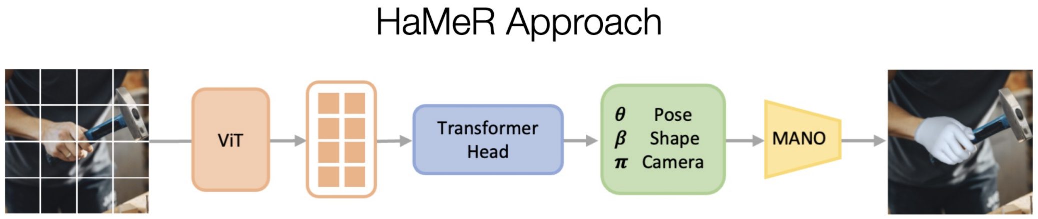


Figure 1: Overview of our method. Given an input video of a human playing piano from a birds-eye view, we use music note transcription to extract piano notes from the audio, and obtain image-aligned 3D hand reconstructions from video. Then, we propose an automated finger labeling method that associates each finger with the closest note, to serve as a dense reward in an RL-based robotic piano playing [1] framework. Our system is able to learn a robotic piano playing policy for a single 15-minutes long piece, including chords and challenging note patterns. We also explored how to use the 3D hand reconstruction as an additional naturalness regularization to improve the naturalness of piano playing.

3.1 Hand Reconstruction from Videos

Given an input birds-eye-view video of a person playing piano, we use HaMeR [4] to reconstruct the 3D hand pose and image-aligned 3D hand mesh at every frame (Fig. 2). HaMeR is a fully transformer-based image-to-hand reconstruction model, and is able to faithfully reconstruct temporally stable hands in a wide variety of scenarios at significantly better quality than prior art.

HaMeR's success lies in scaling both the training data and the network capacity for hand reconstruction: HaMeR is trained with 2.7M examples aggregated from 10 hand datasets, and is built on the vision transformer architecture.



HaMeR uses a fully transformer-based network design. HaMeR takes as input a single image of a hand and predicts the MANO model parameters, which are used to get the 3D hand mesh.

Figure 2: Overview of HaMeR: Reconstructing Hands in 3D with Transformers [4], taken from their paper. Given an input video containing human hands, HaMeR predicts model parameters for the MANO [5] parametric hand model, which can be extracted to retrieve a topology-consistent 3D hand mesh.

3.2 Music Note Transcription from Audio

Given an input audio clip of a person playing piano, we use MT3 [6] to recover the individual music notes played over time (Fig. 3). MT3's outputs are in the standard MIDI format, which represents a musical piece as a sequence of timestamped messages containing "note-on" and "note-off" events. MT3 performs music note transcription from a variety of instruments, including piano, and consistently outperforms prior work on piano note transcription quality from raw music. MT3's success lies in its multi-task training objective, which augments the size of the training data by jointly transcribing arbitrary combinations of musical instruments across several transcription datasets. MT3 is trained with 1749hr of audio aggregated from 6 datasets that contain 14 total instruments.

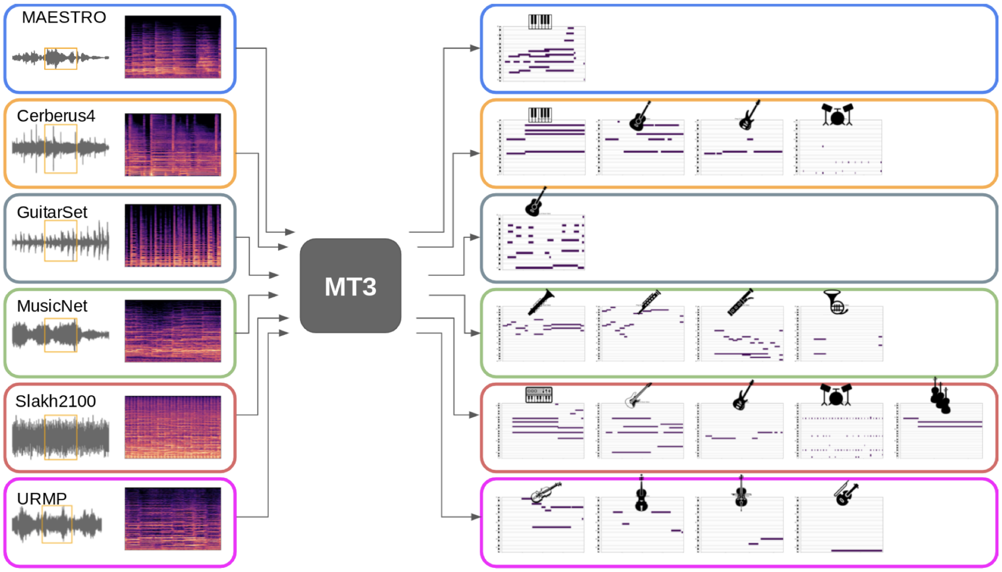


Figure 3: Overview of MT3: Multi-Task Multitrack Music Transcription [6], taken from their paper. Given an input audio clip containing a mixture of instruments, MT3 predicts individual music notes in the standard MIDI format.

3.3 Automatic Fingering Labeling

Given the image-aligned 3D hand reconstruction and corresponding piano note sequence, we propose a method that produces automatic piano fingering labels. Specifically, we compute the location of each fingertip in pixel coordinates by selecting the MANO vertex at the fingertip and projecting its 3D location into the input image. We compute the location of each piano note by taking the center of the corresponding piano key. projecting the 3D location of the fingertip MANO vertex into the input image. Then, each piano note in the MIDI file is assigned to the closest finger from the hand mesh in image pixel space, where the finger-to-key distance is averaged over all frames that the note appears in. The outcome is a piano note sequence with fingering labels in a similar format as the PIG Dataset [3] used by RoboPianist, which we use to achieve dexterous piano playing at scale with in-the-wild birds-eye-view piano playing videos as input.

3.4 Robotic Piano Playing

Given an input piano note sequence with fingering labels, we formulate the robotic piano playing problem as a finite-horizon Markov Decision Process (MDP), defined by a tuple $(\mathcal{S}, \mathcal{A}, \rho, p, r, \gamma, H)$ where $\mathcal{S} \subset \mathbb{R}^n$ is the state space, $\mathcal{A} \subset \mathbb{R}^m$ is the action space, $\rho(\cdot)$ is the initial state distribution, $p(\cdot|s, a)$ are the dynamics, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ are the rewards, $\gamma \in [0, 1)$ is the discount factor, and H is the horizon. The goal of an agent is to maximize the total expected discounted reward over the horizon: $\mathbb{E}\left[\sum_{t=0}^{H} \gamma^t r(s_t, a_t)\right]$.

The observation space contains information about the physical world and the goal state, and includes hand and keyboard joint locations as well as a vector indicating which fingers and piano keys should be in use at each timestep. To enable the agent to plan ahead, the goal state is stacked for some look-ahead horizon L. The action space describes the robot's hand and forearm joint locations. We use the same simulated piano playing environment as RoboPianist [1], which contains a full-size 88-key digital keyboard controlled by two Shadow Hand models from MuJoCo Menagerie [2]. Fig. 4 provides an example of the simulated piano-playing environment.

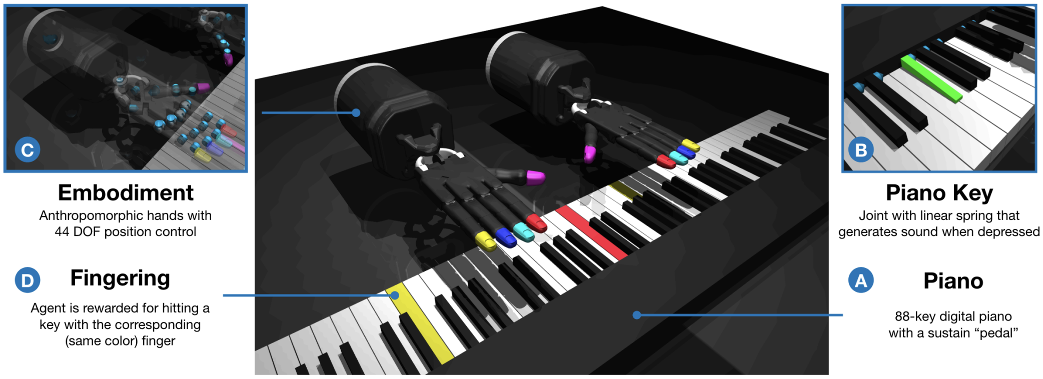

C
Embodiment
Anthropomorphic hands with
44 DOF position control
D
Fingering
Agent is rewarded for hitting a
key with the corresponding
(same color) finger
B
Piano Key
Joint with linear spring that
generates sound when depressed
A
Piano
88-key digital piano
with a sustain "pedal"

Figure 4: Overview of RoboPianist's simulated piano-playing environment, taken from the RoboPianist paper [1]. The environment contains a full-size digital keyboard (A) with 88 piano keys modeled as linear springs (B). Two Shadow Hands (C) are tasked with playing a musical piece, encoded as a trajectory of key presses (D).

4 Results

Below, we provide qualitative results for each step of the automatic fingering labeling pipeline, as well as evaluation metrics for robotic piano playing from videos. All results are on the 14-minute video "Twinkle Twinkle Rousseau", where the piece "Twelve Variations on Ah vous dirai-je, Maman" (written by Mozart) is performed by the YouTube channel Rousseau. We do not evaluate against baselines, as we are not aware of any existing work capable of playing the entire 14-minute piece by watching a human demonstration.

4.1 Hand Reconstruction

We show some representative hand reconstruction results in Fig. 5. The hand reconstruction quality is great from a birds-eye view, as the image-aligned 3D hand meshes almost perfectly match the hand silhouette in the original image. However, the depth of the fingers is often hard to judge from a birds-eye view, due to the shape mismatch between the MANO hand and the actual human hand in the video.

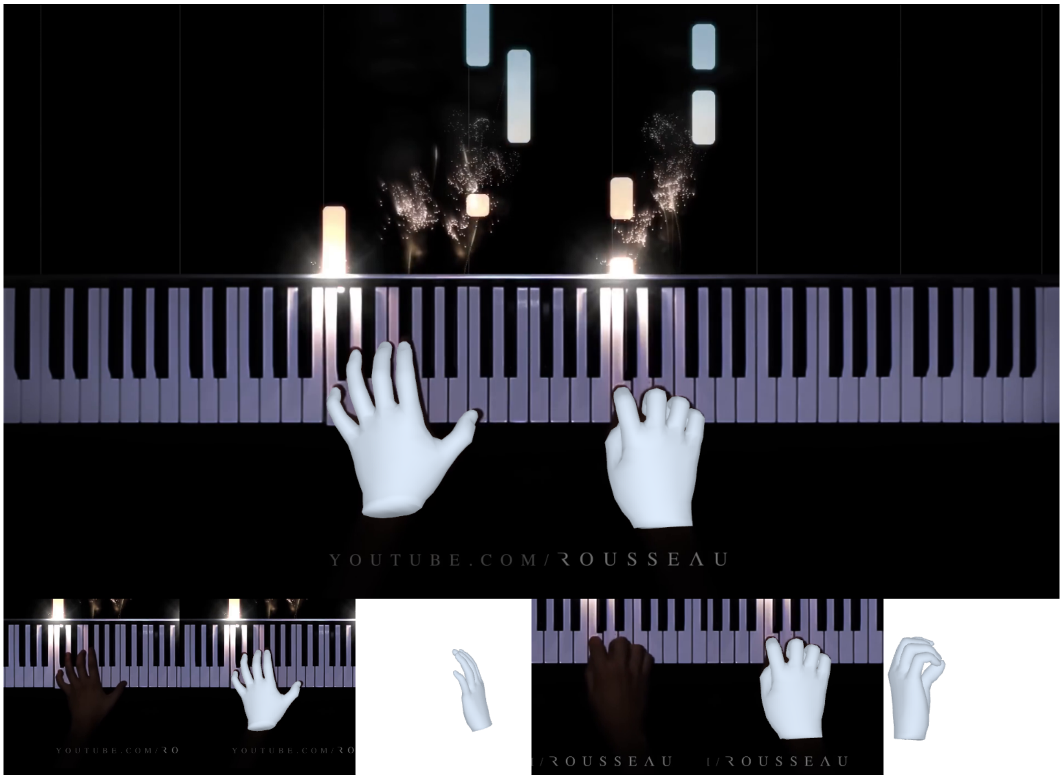


Figure 5: Hand reconstruction examples on Twinkle Twinkle Rousseau. On the top, we visualize the image-aligned 3D hand meshes from a birds-eye view, overlaid on the original video frame. On the bottom, we visualize the left and right hand meshes from a side view. Although the hand meshes are aligned with the input view, the depth of the fingers is inaccurate. For the left hand, only the little finger is actually depressed, even though both the little finger and middle finger are depressed in the hand mesh. For the right hand, only the index finger is actually depressed, even though the thumb is the only depressed finger in the hand mesh.

4.2 Music Note Transcription

We show some representative music note transcription examples in Fig. 6. The transcription quality is good overall, even in challenging regions that contain chords and rapid note sequences. The average error rate is approximately one incorrect note every two seconds.

4.3 Automatic Fingering

We show an example of automatic fingering in Fig. 7. We find that computing the average finger-to-key distance is surprisingly robust, even in more challenging scenarios such as finger-over-finger

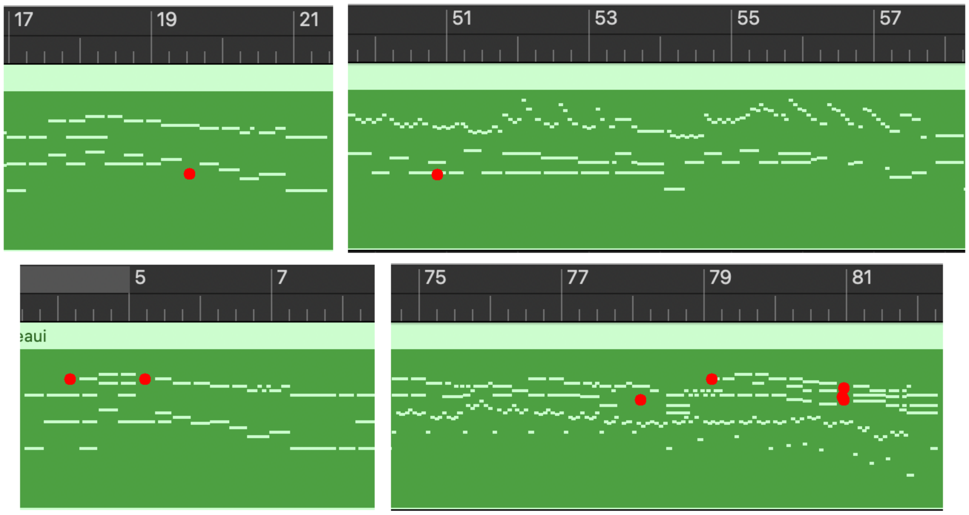


Figure 6: Music transcription examples on Twinkle Twinkle Rousseau. We visualize the MIDI file outputted by our MT3 music note transcription pipeline. Red circles denote either extra or missing notes. The music note transcription quality is good overall, even in challenging regions with chords and rapid note sequences (bottom right). The average error rate is approximately one incorrect note every two seconds.

crossing. However, automatic fingering will fail when either the hand reconstruction or music note transcription fails.

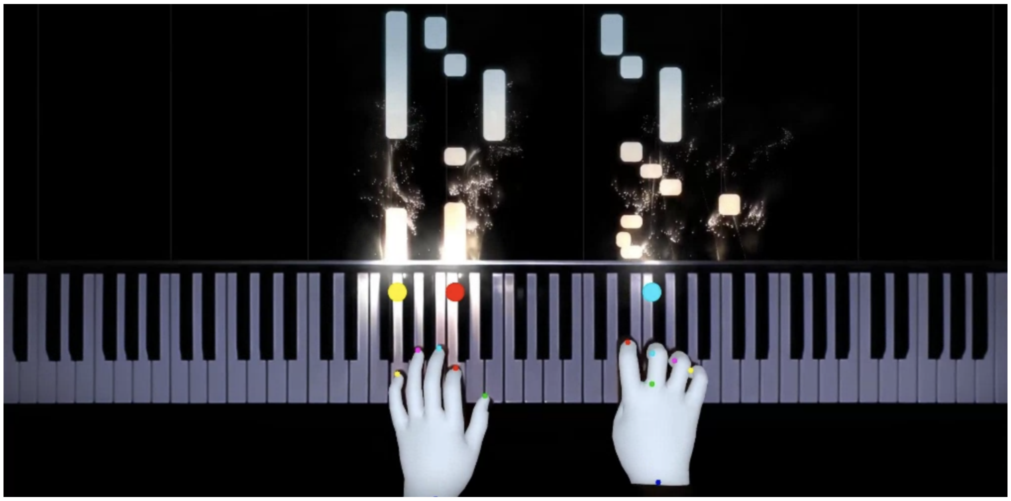
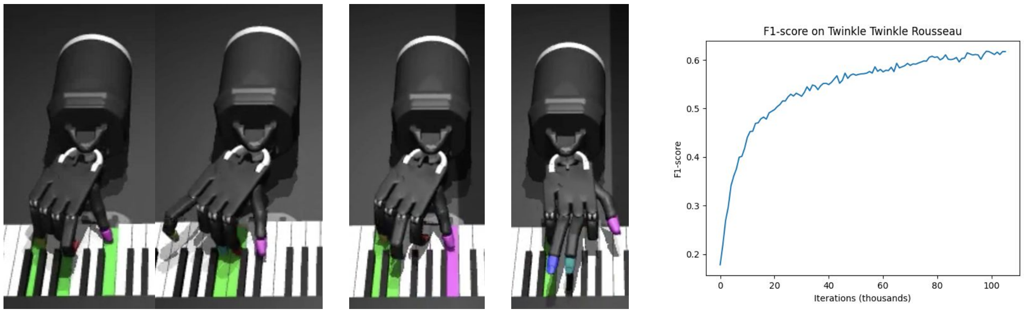


Figure 7: Example of automatic fingering labeling. The small colored circles on each hand denote the estimated location of each fingertip, and the large colored circles on specific keys denote that the corresponding finger should press that key. In this example, the right thumb is in the middle of a finger crossing motion and appears directly underneath the middle finger. The automatic labeling correctly assigns that key to the right middle finger.

4.4 Robotic Piano Playing

We use the F1 score to evaluate the proficiency of our piano playing agent. These metrics compare the state of the piano keys at every time step with the corresponding pseudo-ground-truth state estimated from music note transcription, averaged across all timesteps.

Fig. 8 the F1 score of our piano playing agent throughout the optimization, as well as some examples of expert skills and failure cases. Our method ultimately reaches an F1-score of 0.6 after 100k iterations of optimization, exhibiting skilled piano playing behaviors such as chords (multiple simultaneous keypresses) and trills (two rapidly alternating keys). At the same time, the playing often appears unnatural and some challenging notes can be missed.



(A) (B) (C) (D) (E)

Figure 8: Results from robotic piano playing. Our piano playing agent learns skilled piano playing behaviors, such as the ability to play chords (A) and trills (B). At the same time, the agent sometimes fails to play the correct note (C), even when the dense fingering rewards bring the finger close to the note. Even when the notes are correct, the agent often learns motions that look unnatural to humans (D), such as spurious lifting or pressing motions in the inactive fingers. The F1-score of our method throughout optimization is shown in (E).

5 Conclusion

In summary, we present a method for learning robotic piano playing by watching YouTube videos. In contrast to prior art, our method does not require expert fingering labels as input, and instead learns to imitate the finger motions from a birds-eye-view piano playing video. Our method can learn to play challenging 14-minute-long piano pieces, with chords and challenging fingering patterns.

Limitations. Our method relies on high-quality outputs from two pretrained models: HaMeR for hand reconstruction and MT3 for music note transcription. Though the accuracy of these methods is generally good, they do occasionally fail. Similar to RoboPianist, our method tends to learn an unnatural and mechanical playing style, which lacks much of the fluidity and expressiveness of a skilled human player. Finally, similar to RoboPianist, our method is designed to learn piano playing policies for a specific song, and cannot learn a general piano playing policies to play unseen piano pieces zero-shot.

Future Work. Using the hand reconstructions that our method produces, it would be interesting to investigate how to use human hand priors to promote natural piano playing movements. For example, beyond adding an additional reward for hitting the right key with the right finger, we could align the shape and joints of the shadow hands with those of the reconstructed hand mesh. However, this may be challenging to implement due to limitations in the depth quality of hand reconstructions. Additionally, the strengths of the key presses are not currently reflected by the reward signals and is not considered by the policy, something very important and representative of the quality of human piano playing. This could be an amplitude signal as an additional observation from the audio file to model the strength of each key press. Finally, it would be interesting to learn a generalizable piano playing policy capable of playing unseen piano pieces zero-shot.

References

[1] K. Zakka, P. Wu, L. Smith, N. Gileadi, T. Howell, X. B. Peng, S. Singh, Y. Tassa, P. Florence, A. Zeng, et al. Robopianist: Dexterous piano playing with deep reinforcement learning. In *7th Annual Conference on Robot Learning*, 2023.

[2] K. Zakka, Y. Tassa, and M. M. Contributors. Mujoco menagerie: A collection of high-quality simulation models for mujoco. URL http://github.com/deepmind/mujoco_menagerie.

[3] E. Nakamura, Y. Saito, and K. Yoshii. Statistical learning and estimation of piano fingering. *Information Sciences*, 517:68–85, 2020.

[4] G. Pavlakos, D. Shan, I. Radosavovic, A. Kanazawa, D. Fouhey, and J. Malik. Reconstructing hands in 3d with transformers. *arXiv preprint arXiv:2312.05251*, 2023.

[5] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *SIGGRAPH ASIA*, 2017.

[6] J. Gardner, I. Simon, E. Manilow, C. Hawthorne, and J. Engel. Mt3: Multi-task multitrack music transcription. *arXiv preprint arXiv:2111.03017*, 2021.