Expressive Piano Playing Robotic Finger Control

Yuyi Zhang

Abstract

By replicating the MIDI signal sequence played by a human pianist, we made robot achieve expressive piano playing beyond the sheet music. We collected thousands of pressing data and mapped the MIDI message to control parameters using smoothing spline, Gaussian process, Densely Connected Neural Network and LSTM. The expressive playing control system we developed aims to mimic single-finger expressive human piano playing by pressing piano keys with correct velocity and timing.

1 Background

1.1 MIDI: Musical Instrument Digital Interface

In digital pianos, Musical Instrument Digital Interface(MIDI) acts as a widely agreed data communication protocol which defines the information exchange between speaker control signals and the piano instrument [1]. Printed piano music stores the symbolic instruction of musical notes, but still needs a human to read and interpret the message. MIDI, however, contains all essential messages for the electrical circuit to recreate a music [1] in a format of:

```
[[[144, note, pressing velocity, 0], pressing timestamp],
[[128, note, releasing velocity, 0], releasing timestamp]]
```

The digital pianos are able to capture a pianist's expression and interpretation of a music piece in its MIDI message sequence generated from the human playing. A straightforward method to give robot the appearance of human expressiveness is mimicry: we record the MIDI sequence from a human playing piece, which includes each note's length, pitch, and key-strike velocity, and control the robot to transcript the human performance by producing the same MIDI sequence [2].

1.2 Piano playing robots

Previous researches about piano playing robots have two major focuses: the mechanical actuation of the fingers and the finger movement planning algorithms [3]. Most robots play the piano according to the music sheets, either by reading the printed scores using a camera system [4], or by designating notes in codes [5]. Scimeca et al. [3] have analysed the dynamic interactions between a stiff finger and the piano for expressive single key pressing, but it's still unclear how the robot can play a sequence of notes expressively, or whether the expressive playing control approach would work for a soft robotic finger.

1.3 Soft finger and particle jamming stiffness control

Soft fingers made of flexible materials offer more flexibility, conformability, and safety than rigid ones. Pneumatic particle jamming allows us to modulate the soft finger's stiffness by controlling the vaccum pressure inside the particle capsule [6]. A particle jammed soft finger's variable stiffness and soft texture can potentially parody a human hand's performance in piano playing by displaying more flexibility and variation in its interaction with the piano key. In this project, we will investigate the potential of a particle jammed soft finger in performing expressive piano playing.

1.4 Data-driven robot control

To make the robot perform expressive piano playing, it is crucial to design a proper control technique for the robotic finger. Data-driven methods have been widely applied in non-linear problems that is tricky for conventional physical model to explain. Using Gaussian Process, Scimeca et al. [3] achieved 10 different playing styles by mapping the non-linear relationship between MIDI signal and UR5 robot control parameters. By training an artificial neural network (ANN) to mimic a real flutist's vibrato and note duration, Solis et al. [7] achieved a significant gain in expressiveness of flutist robot's performance. The soft finger's characteristic and the piano's unknown internal structure make non-linearity in the robot-piano system non negligible in this project. To describe the connections between control parameters and MIDI messages, we will use data-driven methods such as neural networks and Gaussian processes to build the soft finger control model.

Piano music can be represented by a sequence of MIDI messages. The corresponding control parameters for playing such a MIDI sequence will thus be sequential too. Recurrent Neural Network (RNN) and Long Short-Term Memory Network (LSTM) have shown great improvement over other neural network structures on processing temporal, sequential data [8]. To make a control system capable of dealing with sequential MIDI, we will use RNN and LSTM.

2 Experimental set-up

For the experiment we use UR5 Robot arm with a passive finger attached to the end of it(Fig 1a). The Tool Center Point(TCP) coordinate of UR5 is set to be at the fingertip. By sending the control command with specified acceleration, velocity and distance to UR5, the UR5 TCP will move trapezoidally, which causes the attached finger to press the key on digital piano Kawai ES8. The piano will generate MIDI message and play the sound accordingly. The laptop connected to UR5 and ES8 piano will record control parameters and MIDI messages for each key pressing.

2.1 Finger design

The rigid finger is a cylindrical $80mm \times 15mm$ attachment with a flat origin and a rounded end(Fig 1c). The finger was 3D-printed with FilaFlex, a Thermoplastic Polyether-Polyurethane

Elastomer(TPE) filament with a shore hardness of 82A, which allows some flexing and bending [3].

The soft finger is made from a 3D printed rigid skeleton of index finger wrapped in a silica gel shell. The skeleton does not deform in key pressing. The hollow capsules between bones and the silica gel shell provides the finger with some flexibility for deformation in the interaction with the piano. The finger joints are filled with ground coffee and connected to a compressor by tubes, a pneumatic regulator and an inverter(Fig 1b). The vacuum pressure (-80∼0 kPa) inside finger can be changed by tuning the pneumatic regulator to achieve various stiffness of the soft finger.

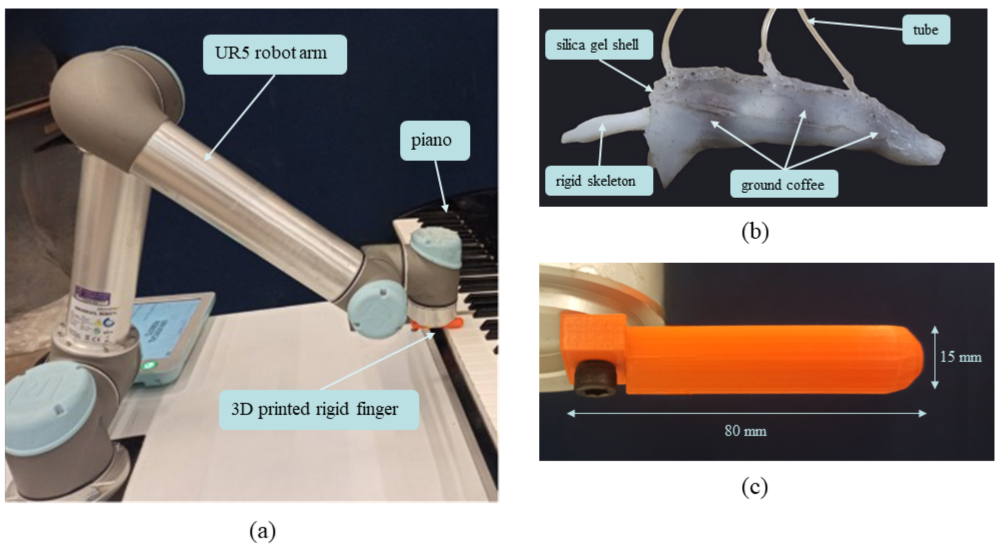

Figure 1: The set-up for the experiments. (a) shows robotics set up, including a schematic of the robot connection to a processing unit and the digital piano. (b) the soft finger with particle jamming. (c) shows the rigid 3D finger used for piano playing.

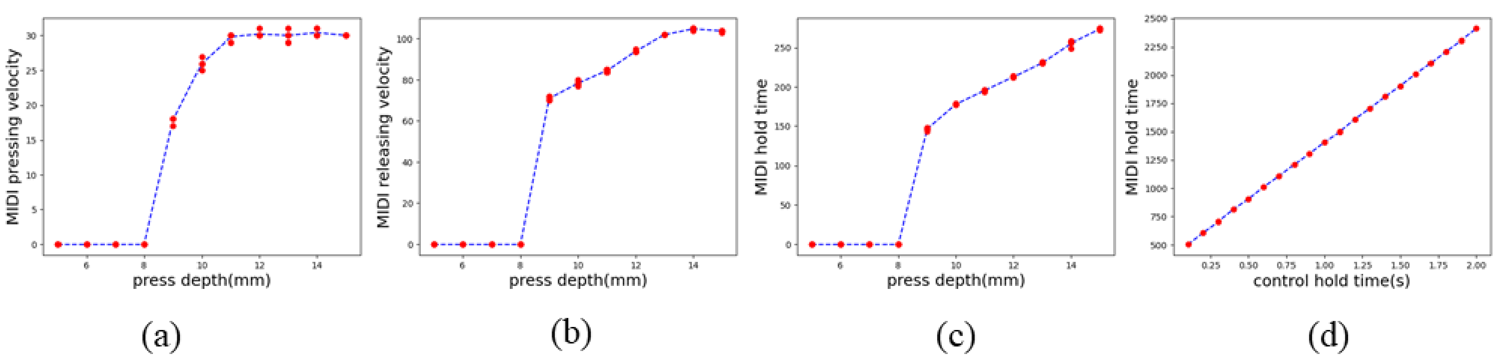

Figure 2: Pre-experiment results: (a,b,c) rigid finger experiments with changing pressing depth and fixed finger angle 0°, UR5 down velocity $0.1m/s$, UR5 up velocity $0.1m/s$, key-finger distance $0.1m$, hold time $0s$, UR5 acceleration $1m/s^2$ (a) MIDI pressing velocity to UR5 pressing depth; (b) MIDI releasing velocity to UR5 pressing depth; (c) MIDI hold time to UR5 pressing depth; (d) MIDI hold time to UR5 hold time

3 Progress to date

3.1 Pre-experiment

Control parameter	Rigid finger	Soft finger
UR5 down velocity	$0.01 \sim 0.5\ m/s$	$0 \sim 0.06\ m/s$
UR5 up velocity	$0.01 \sim 0.7\ m/s$	$0 \sim 0.7\ m/s$
UR5 acceleration	$0.05 \sim 2\ m/s^2$	$0.05 \sim 2\ m/s^2$
Key-finger distance	$0 \sim 0.05\ m$	$0 \sim 0.05\ m$
Key-finger angle	$0 \sim 90°$	$0 \sim 90°$
UR5 pressing depth	$0.008 \sim 0.015\ m$	$0.020 \sim 0.035\ m$
UR5 hold time	$0 \sim 2s$	$0 \sim 2s$
Vacuum pressure	N/A	$-80 \sim 0kPa$

Table 1: Proper control parameter ranges located in pre-experiments. In future data collection, we will use control parameters within these ranges.

We did pre-experiments by changing one control parameter and fixing the others. Sample data in Figure 2 shows the non-linearity between pressing depth and MIDI message, while the UR5 hold time is strictly linear to the MIDI hold time (gap between pressing and releasing timestamp). Combining the pre-experiment results and the data collection experience, we located the proper range of control parameters in which the major changes of MIDI message happens (Table 1).

3.2 Data collection

We performed three rounds of data collection:

Round No.	1	2	3
Finger type	rigid	soft	rigid
Sequence length	1	1	12
UR5 velocity (m/s)	$0.05 \sim 0.5$	$0 \sim 0.06$	$0.01 \sim 1.0$
UR5 acceleration (m/s^2)	10	5	$0.1 \sim 2.0$
Key-finger distance (m)	0.3	0	$0 \sim 0.08$
Key-finger angle	0°	0°	0°
UR5 pressing depth (m)	0.02	0.035	$0.009 \sim 0.015$
UR5 hold time (s)	0	0	$0 \sim 3$
Vacuum pressure (kPa)	N/A	$-80 \sim 0$	N/A
Pressing number	95	250	1151

Table 2: Control parameter settings for three data collection rounds. For the range value, we use grid searching or uniform random draw to decide the control parameter value for each pressing. Vacuum pressure indicates the stiffness of particle jammed soft finger and thus is non applicable to rigid finger.

Round 1, we recorded 95 (control parameters, MIDI) separate key pressings pairs performed by rigid finger. We set the UR5 down and up velocity to be the same, uniformly distributed in the range [0.05,0.5] m/s with an interval of 0.5 m/s. All other UR5 control parameters are fixed. We sent a pressing control command to UR5 and recorded the generated MIDI.
Round 2, we recorded 250 (control parameters, MIDI) separate key pressings pairs by particle jammed soft finger with vacuum pressure [-80,-60,-40,-20,0] kPa using the same procedure as in round 1.
Round 3, we recorded 100 sequences of 12 (control parameters, MIDI) pairs with uniformly drawn UR5 control parameters from the range shown in table 2. We ended up with 1151 pressings due to interruptions from the UR5 robot runtime disconnection.

3.3 Data modeling

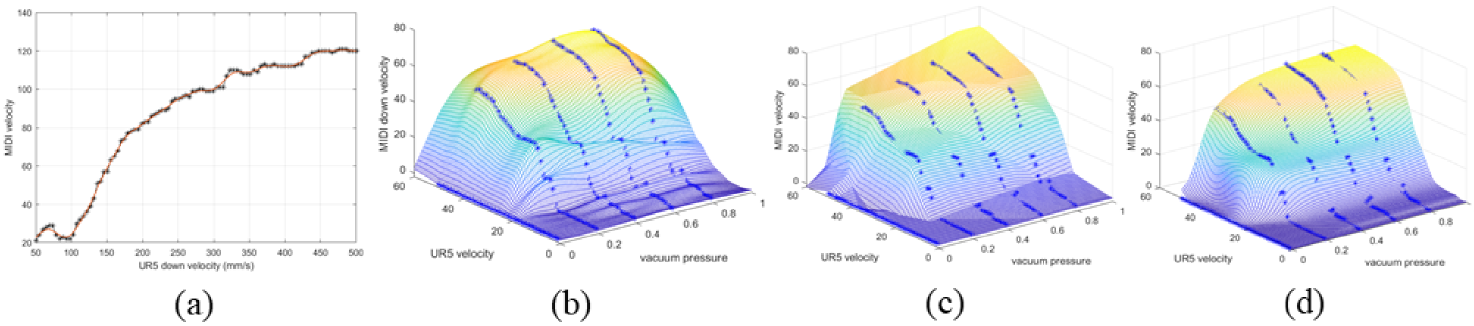

Figure 3: Modelling results: (a) rigid finger UR5 pressing velocity to MIDI pressing velocity; (b)(c)(d) soft finger vacuum pressure and UR5 pressing velocity to MIDI pressing velocity with (b) Gaussian Process (c) Non optimized densely connected neural network (d) Bayesian optimized densely connected neural network

3.3.1 Rigid finger single note - data collected in round 1

Plotting UR5 down velocity in x-axis and MIDI pressing velocity in y-axis, we used smoothing spline to fit the relationship between them (Fig 3a). The range of possible MIDI pressing velocity is 0$\sim$127, the rigid finger can almost achieve maximum MIDI velocity by accelerating to 0.5 m/s. Specifying a desired MIDI pressing velocity, the required UR5 down velocity can be looked up from the smoothing spline model. When there are multiple choices of UR5 down velocity for a MIDI, we always choose the smaller one in the demonstration session.

3.3.2 Soft finger single note - data collected in round 2

For soft finger data collected in round 2, we used [vaccuum pressure, UR5 down velocity] as input $\mathbf{x}$ and MIDI pressing velocity as output y. We used Gaussian Process (Fig 3b), Densely Connect Neural Network (DCNN) (Fig 3c,3d) to build a regression model from $\mathbf{x}$ to y. To gain the require UR5 down velocity for a MIDI under some value of vacuum pressure, we can look up the plotted regression model.

Model	Structure	Evaluate
Gaussian Process	$CovFunc = covSEard_1 + covSEard_2$	-loglikelihood=469.17
DCNN	[10 10], relu, linear	test loss=6.8415
Optimized DCNN	[3 291], tanh, linear	test loss=1.843

Table 3: Model performances on the soft finger. The structure column shows convariance function for Gaussian Process and fully connected layer size, activation function, output activation function for the Densely connected neural network

3.3.3 Rigid finger sequential notes - data collected in round 3

For rigid finger sequential data collected in round 3, we used the MIDI sequence as input and control parameter sequence as output to train RNN and LSTM models. The sequential data were split into training and testing data in a ratio of 8:2. The RNN has 2 layers, hidden dimension 12 and yields testing loss 0.6683 after 300 epochs with learning rate 0.01. The LSTM has 2 layers, hidden dimension 24 and yields testing loss 0.5732 after 100 epochs with learning rate 0.001.

4 Future work

4.1 Non-linearity in soft finger's interaction with piano

The next stage of the project will focus on the non-linearity of particle jammed soft finger when it interacts with the piano. By applying 3D camera tracking system, we will record fingertip velocity in the process of finger hitting the piano. The UR5 velocity sequences will then be related with the fingertip velocity sequences using RNN or LSTM model, which will potentially reveal the internal non-linearity of the soft finger.

4.2 Soft finger sequential notes

We also plan to perform the sequential data collection on the pneumatic stiffness controlled soft finger. We will explore whether the current sequential control parameter prediction model for rigid finger will work for the soft finger too.

4.3 Comparison to human piano playing

By recording MIDI sequence from human playing piano with one finger, we create a expressive music piece. We will use the trained RNN or LSTM model to predict the control parameters needed for UR5 and soft finger to replicate the human playing MIDI. We then compare the music played by human and by the robot in terms of MIDI velocities, note length and timing. Candidates will be ask to try distinguishing these two. The combined result will reflect the quality of our expressive piano playing control system.

References

[1] Joseph Rothstein. *MIDI: A comprehensive introduction*, volume 7. AR Editions, Inc., 1992.

[2] Angelica Lim, Takeshi Mizumoto, Toru Takahashi, Tetsuya Ogata, and Hiroshi G Okuno. Programming by playing and approaches for expressive robot performances. In *IROS Workshop on Robots and Musical Expressions*, 2010.

[3] Luca Scimeca, Cheryn Ng, and Fumiya Iida. Gaussian process inference modelling of dynamic robot control for expressive piano playing. *Plos one*, 15(8):e0237826, 2020.

[4] Ichiro Kato, Sadamu Ohteru, Katsuhiko Shirai, Toshiaki Matsushima, Seinosuke Narita, Shigeki Sugano, Tetsunori Kobayashi, and Eizo Fujisawa. The robot musician 'wabot-2' (waseda robot-2). *Robotics*, 3(2):143–155, 1987. Special Issue: Sensors.

[5] Dan Zhang, Jianhe Lei, Beizhi Li, Daniel Lau, and Craig Cameron. Design and analysis of a piano playing robot. In *2009 International Conference on Information and Automation*, pages 757–761. IEEE, 2009.

[6] Yang Yang, Yingtian Li, and Yonghua Chen. Principles and methods for stiffness modulation in soft robot design and development. *Bio-Design and Manufacturing*, 1(1):14–25, 2018.

[7] Jorge Solis, Kei Suefuji, Koichi Taniguchi, Takeshi Ninomiya, Maki Maeda, and Atsuo Takanishi. Implementation of expressive performance rules on the wf-4riii by modeling a professional flutist performance using nn. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2552–2557. IEEE, 2007.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.