Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

YQX+: Benchmarking Probabilistic Modeling of Piano Performance Expression

Jinwen Zhou

jinwen.zhou@qmul.ac.uk

Queen Mary University of London

Yuncong Xie

New York University

Haochen Wang

New York University

Jinhua Liang

Queen Mary University of London

Huan Zhang

Queen Mary University of London

Simon Dixon

Queen Mary University of London

Aidan O. T. Hogg

Queen Mary University of London

Research Article

Keywords: Benchmarking, Expressive Performance Evaluation

Posted Date: November 12th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11241882/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

YQX+: Benchmarking Probabilistic Modeling of Piano Performance Expression

Jinwen Zhou[1*], Yuncong Xie[2], Haochen Wang[2], Jinhua Liang[1], Huan Zhang[1], Simon Dixon[1], Aidan O. T. Hogg[1]

[1*]Centre for Digital Music, Queen Mary University of London, London, UK.

[2]Music Technology, MPAP, Steinhardt, New York University, New York, NY, USA.

*Corresponding author(s). E-mail(s): jinwen.zhou@qmul.ac.uk;
Contributing authors: yx2568@nyu.edu; hw2346@nyu.edu;
jinhua.liang@qmul.ac.uk; huan.zhang@qmul.ac.uk;
s.e.dixon@qmul.ac.uk; a.hogg@qmul.ac.uk;

Abstract

A written music score specifies what notes should be played, but much of the musical meaning of a performance comes from how those notes are interpreted: pianists shape timing, dynamics, tempo, and articulation to communicate structure, style, and emotion. Modeling this expressive aspect of music matters for music education, digital performance rendering, and creative AI systems, and is difficult precisely because the same score admits many different but musically valid interpretations.

In this paper, we present YQX+, a benchmarking framework for probabilistic modeling of expressive piano performance. YQX+ shifts the focus from demonstrating a single rendering system to comparing how different models capture the expressive variation observed in human performance. To make that comparison possible, it fixes the prediction task, a shared multi-scale contextual feature representation, and a common evaluation protocol, so that model families are trained and compared on equal footing. Given a symbolic score, every model in the suite predicts the same note-level expressive parameters, including beat period, velocity, timing deviation, and articulation ratio, from features that capture local note context, harmonic texture, phrase structure, and score markings.

The suite takes the original YQX Gaussian mixture model (GMM) as its reference point and extends it to XGBoost, variational autoencoders, and conditional flow-matching models.
To support perceptual evaluation, we also introduce PerformanceMOS, a data-driven estimator of perceived expressive quality and performance skill from symbolic piano performances. Experiments show that larger-capacity XGBoost and β-VAE models achieve lower timing and articulation errors than prior baselines, including DExter and VirtuosoNet, and obtain the highest predicted mean opinion scores (MOS). These findings establish YQX+ as a systematic benchmark for probabilistic modeling of piano performance expression, with PerformanceMOS providing a scalable proxy for human listening evaluation. The code, trained models, and evaluation scripts are available at `https://github.com/kkwsts/YQX-Plus`.

Keywords: Benchmarking, Expressive Performance Evaluation

1 Introduction

A written piano score specifies the notes to be played, but it does not fully determine how the music should sound. In Western classical piano performance, performers shape the score through nuanced choices of timing, dynamics, tempo, and articulation, allowing the same piece to give rise to different but musically valid interpretations. This expressive component is central to a performance, as it conveys phrasing, stylistic character, and the performer's interpretative intent.

Computationally modeling this expressive component remains a longstanding challenge in music information retrieval (MIR), with applications in music education, digital performance rendering, and creative AI systems [1–3]. The challenge is not only to generate a plausible performance, but also to model the range of expressive choices that human performers may make from the same score. This motivates probabilistic approaches, which can represent performance expression as a distribution of possible interpretations rather than as a single deterministic output.

The original YQX system [1], winner of the 2008 Automatic Performance Rendering Contest (RenCon) [4], provided an early demonstration of this idea by learning

expressive control parameters such as local tempo, loudness, and articulation from human piano performances. However, YQX was centered on a Gaussian mixture model approach and was not designed as a modular benchmark framework for comparing modern probabilistic and generative models. More recent systems, including diffusion-based expressive control models [5] and multimodal rendering architectures such as RenderBox [6], have advanced expressive performance generation. Yet these systems are difficult to compare directly, because they often differ in their score representations, output targets, modeling assumptions, and evaluation protocols.

This paper presents YQX+, a benchmarking framework built around a modern reimplementation of the original YQX system. YQX+ revisits score-conditioned performance prediction as a supervised learning task: given a symbolic score, the model predicts note-level expressive parameters describing how the score is performed. The framework combines structured musical features, a unified target representation, systematic feature ablations, and a perceptually grounded evaluation.

Figure 1 illustrates this score-conditioned prediction task, showing how a notated score is aligned with a performed version and how note-level expressive targets such as beat period, dynamics, timing, and articulation are extracted.

This evaluation component is particularly important because reconstruction-based objective errors (such as deviation) in timing, dynamics, or articulation do not necessarily capture whether a rendered performance sounds natural, expressive, or skilled to listeners. To address this, we introduce PerformanceMOS, a data-driven estimator of perceived performance quality from symbolic piano performances.

As a result, YQX+ shifts the focus from demonstrating a single rendering system to comparing how different models capture the expressive variation observed in human performance.

The main contributions of this work are as follows:

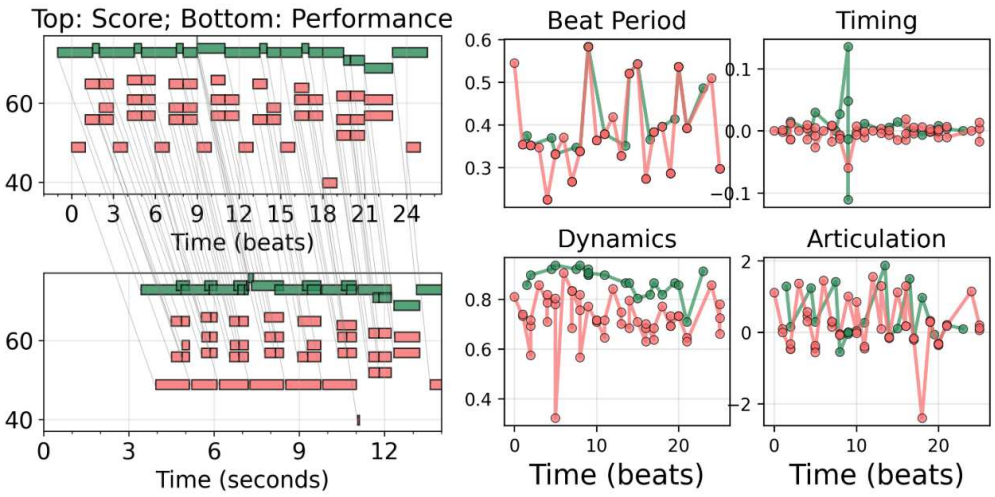


Fig. 1 The opening phrase of Schubert's *German Dance D783, No. 15*, with top voice in green and bottom voice in red. The left panels show the piano-roll representations of the score (top) and performance (bottom), with alignment of the symbolic note representations in beats and seconds respectively. The right panels show four expressive targets extracted from the snippet.

- We introduce YQX+, a unified score-conditioned expressive performance prediction framework that extends the original YQX system with richer musical representations and benchmarking capabilities for modern probabilistic modeling approaches.
- We evaluate a range of models within this framework, from traditional machine learning methods to deep generative approaches, including Gaussian mixture models, XGBoost, Variational Autoencoders, and flow-matching networks. Through systematic feature ablations, we examine the relative importance of different levels of score context for predicting expressive performance parameters.
- We introduce PerformanceMOS, a data-driven model for estimating perceived performance quality, including skill, fluency, and expressiveness, from symbolic piano performances. This provides a scalable perceptual evaluation component for comparing YQX+ models and existing expressive rendering baselines.

The remainder of this paper is organized as follows. Section 2 reviews related work on expressive performance rendering and perceptual evaluation. Section 3 introduces the YQX+ representation framework, including the score-derived input

features, expressive performance targets, and probabilistic modeling approaches. Section 4 describes the datasets, feature configurations, and training setup. Section 5 presents the objective metrics, PerformanceMOS evaluation, and subjective listening study. Section 6 discusses the findings and outlines future extensions of the YQX+ framework.

Before presenting these sections, we introduce the key musical terminology. **Music Score**: A written guide in Western notation showing idealized pitch, timing, and markings for performance. **Performance**: A symbolic representation of performed musical events with absolute timing. **Music Note**: A musical event indicating the pitch, start time, and duration of a sound. **Joint Onset**: A group of notes notated as starting simultaneously. **Voice**: A sequence of notes that is perceived as a coherent musical line.

2 Background and Related Work

This section reviews two areas of prior work that motivate YQX+. First, we consider expressive performance rendering, focusing on how existing systems represent the score, model the mapping from score to performance, and define the expressive parameters to be predicted. Second, we review perceptual evaluation methods, since benchmarking expressive performance models requires not only objective error metrics, but also ways to assess whether generated performances sound natural, expressive, and skilled to listeners.

2.1 Expressive Rendering

Expressive performance rendering has long been studied in music information retrieval, with early work recognizing the role of machine learning [7] alongside several rule-based approaches [3, 8, 9]. In this section, we review learning-based approaches according

Table 1 Summary of expressive performance rendering models according to their input features, modeling approach, and target representation. Acronyms used in the table: IR, Narmour's implication-realization model; IOI, inter-onset interval; GMM, Gaussian mixture model; CRF, conditional random field; RNN, recurrent neural network; LSTM, long short-term memory network; FFNN, feedforward neural network; VAE, variational autoencoder; SSL, self-supervised learning; XGBoost, extreme gradient boosting; CFG, classifier-free guidance; CFM, conditional flow matching.

Method	**Year**	**Features**	**Modeling**	**Target**
YQX	2009	7 mid-level features (pitch interval, rhythmic context, IR, etc.)	GMM	IOI formulation
Polyhymnia	2011	Piano-roll + one-hot vectors of markings	CRF	IOI formulation
Basis Mixer	2018	Collection of basis functions: pitch, dynamics markings, piano-roll, harmonic tension, segmentation features, etc.	RNNs/LSTMs (onset-wise); FFNNs (note-wise)	Basis Mixer style
VirtuosoNet	2019	Hierarchical score features in note/beat/measure levels	Hierarchical RNN with multi-level attention	Tempo, velocity, onset deviation, articulation and pedal usages
Maezawa et al.	2019	7-beat local piano-roll + one-hot encodings (pitch, IOI, duration, tempo, velocity)	Conditional Variational RNN	Velocity, onset deviation, duration, and tempo fluctuation (categorical)
SketchExpression	2022	8 low-level score features (pitch, duration, chord position, etc.)	Conditional VAE with SSL	IOI formulation
midihum	2023	∼400 engineered features from seq. modeling (pitch oscillators)	XGBoost	Velocity only
ScorePerformer	2023	SPMuple: 8 score tokens + 4 performance tokens (tokenized tuples)	Transformer w. VAE encoding heads	SPMuple tokens
DExter	2024	Score codec ($4 \times n$ matrix) + Perceptual codec ($7 \times n$), note level	Diffusion w. CFG conditioning	Basis Mixer style
PianoFlow	2025	Score feature (pitch, position, PositionShift, duration)	CFM	Performance features (velocity, TimeShift, TimeDuration, TimeDurationSustain)

to three aspects: the score-derived information provided to the system, the modeling approach used to learn the mapping from score to performance, and the target representation used to capture and reconstruct expressive detail.

For learning-based expressive performance rendering, these three aspects provide a useful basis for comparison, because existing systems differ not only in their model architectures but also in the musical information they use as input and in how they represent expressive output. YQX [1], the direct predecessor of the present framework, illustrates an early instance of this design space along three aspects. Its features are a set of seven descriptors, pitch interval, rhythmic context, and measures derived from Narmour's implication-realization (IR) theory [10, 11]. The modeling approach is a simple Bayesian network, in which continuous features are treated as Gaussian-distributed and discrete features as probability tables, with each target predicted by linear regression over a joint distribution estimated separately for every combination of discrete feature values. The target representation is the inter-onset interval (IOI) ratio together with loudness and articulation, computed for a single monophonic voice, with the remaining voices synchronized to it afterward.

Polyhymnia [12] keeps YQX's IOI-style target formulation, but replaces the Bayesian network with a Conditional Random Field (CRF) over a sequence of score feature vectors, letting sequential dependency between notes be modeled directly in the graphical structure rather than imposed afterward. The distinctive strategy is structural rather than learned, where each chord is split into a 'parent' note and 'additional' (decorative or doubling) notes, with the additional notes' loudness modeled as dependent on their parent's rather than predicted independently.

Basis Mixer [13] extends this design. The features generalize YQX's fixed set into an open-ended collection of basis function that uniformly encode a wider range of score information (i.e. dynamic markings). While the framework originated from closed-form linear regression, its modeling approach evolved into a pair of non-linear sequence models. The onset-wise targets are modeled with recurrent networks (RNNs/LSTMs) and gated recurrent units (GRUs), while note-wise targets are modeled with feed-forward networks (FFNNs) over local context. The target representation decomposes

note-level timing into beat period, representing global tempo, and timing deviation, representing local expression.

VirtuosoNet [2] replaces hand-designed higher-level context features (i.e. phrase) with a fully learned hierarchical score encoder, where the beat- or measure-level context is pooled from note-level representations via attention rather than computed analytically. The hierarchical RNN-based model generates tempo and dynamics at the measure level first, and refine them into note-level tempo, velocity, onset deviation,articulation, and pedal use.

For more recent work, DExter [5] encodes performances in a fully invertible, Basis Mixer style note-level codec, conditioned on a score codec and a perceptual codec for steerable generation. A diffusion model is used for capturing global tempo and dynamics trajectories. PianoFlow [14] uses four score features (pitch, note's position inside the bar, PositionShift for an inter-note-interval between consecutive score notes, and duration), embedded continuously rather than as discrete tokens to preserve relative pitch and timing information, and trains a masked conditional flow matching (CFM) model that learns to inpaint whichever performance features a sampled mask hides. Its target representation is four performance features per note, velocity, TimeShift (merges tempo and onset deviation into a inter-onset local-tempo measure), TimeDuration and TimeDurationSustain (a pedal-aware note duration split into pressed and sustained components, jointly approximate articulation and pedal use).

Table 1 summarizes these differences across prior symbolic expressive rendering systems. Several representative systems illustrate why such comparisons are difficult. Maezawa et al. [15] and Jeong et al. [2], for example, use sequential models such as recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), but differ in the score context provided to the model and in the expressive parameters they predict. More recent controllable rendering approaches [16, 17] introduce conditioning mechanisms through variational models, hierarchical style encoders, or

token-based control representations. While these systems improve generation and control, their differing inputs, targets, and modeling assumptions make direct comparison across methods difficult.

Target representations also vary substantially across prior work. Many transformer-based models [18, 19] output performed notes directly as symbolic tokens, which does not explicitly separate score content from expressive deviations. One early representation that explicitly models expressive timing is the IOI formulation [1, 12], which represents timing through inter-onset intervals between adjacent notes. However, this formulation assumes a one-dimensional note stream, making it less suitable for polyphonic piano music. Basis Mixer [13] decomposes note-level timing, generalizing to polyphonic textures in a way IOI formulation does not. The target representation used by PianoFlow [14] differs from the others reviewed here in combining what they treat as separate targets into fewer.

2.2 Performance Evaluation

Evaluating expressive performance rendering is likewise difficult. Objective evaluation is almost always reconstruction-based: the predicted expressive parameters are compared with those extracted from one recorded human performance of the same score, and the resulting differences in timing, dynamics, or articulation are reported as errors. The term is inherited from supervised learning, where any departure from the held-out target counts as an error, but a score admits many valid interpretations, so a difference from one particular reference performance is not in itself a defect. Peter et al. [20] examine this assumption directly: in a listening test they find that listeners perceive subtle differences between performances that quantitative reconstruction measures do not capture, and in further experiments they find that the reliability and validity of such measures vary considerably across reference performances and pieces. We therefore treat the deviation measures reported in Section 5 as differences from one specific

reference performance rather than as errors in an absolute sense, and pair them with perceptual measures. Human listening tests provide the perceptually meaningful judgments that reconstruction measures lack, but they are costly, time-consuming, and difficult to scale.

A complementary line of work assesses a performance directly, without comparing it to a matched recording of the same piece. Music performance analysis has a long history of relating measurable properties of a performance to human judgments of its quality [21]. One strand of recent work predicts those judgments. Expert and novice listeners have been shown to invoke systematically different criteria when assessing the same student performances [22], which makes the provenance of the ratings part of the evaluation design rather than an incidental detail. Datasets pairing performances with expert annotation have made rating prediction tractable: PercePiano [23] labels 1,202 performance segments with 19 perceptual features grouped into eight categories, while the NeuroPiano [24] and YCU-PPE [25] corpora provide examiner ratings of student performances. Models trained on such data range from feature-based regressors to language-model coaches that answer free-form questions about a performance [26].

A second strand localizes what a listener notices as wrong, rather than scoring the performance as a whole. Morsi et al. [27] detect conspicuous mistakes without access to the score, on the observation that some errors are salient to listeners while others pass unnoticed. Subsequent work [28] organizes these phenomena into a hierarchical taxonomy of piano performance mistakes, in which a low level of granular score deviations (pitch insertion and deletion, note delay and anticipation, note extension and shortening, and time shifts or jumps within the score) composes into a mid level of core mistakes and recovery behaviours such as mistouches, substitutions, time interruptions, dragging, and note repetitions, each associated with the musical texture in which it occurs. Because the taxonomy is generative as well as descriptive, it can be used to synthesize mistakes on mistake-free performances, and it has since been used to

augment score-aligned corpora for expressive rendering [6]. RUMAA [29] unifies score-performance alignment, score-informed transcription, and mistake detection within a single model. This strand is complementary to the present work: it identifies departures from the notated score that are unintended, whereas the expressive parameters modeled here are precisely the departures that are intended.

Beyond the music domain, this problem is related to automatic quality assessment in speech, music, and audio, where data-driven models are increasingly used to approximate human perceptual ratings [30–32].

In speech, perceptual quality has traditionally been evaluated using human listening tests, particularly mean opinion score (MOS) evaluations. The VoiceMOS Challenge series [33], initially launched in 2022, has encouraged the development of reference-free (non-intrusive) MOS prediction methods that estimate human ratings from data. Similar ideas have also been extended to music and general audio evaluation [34]. For example, Audiobox Aesthetics [30] and SongEval [31] use human-annotated perceptual data to predict qualities such as musicality, structure, and production quality.

However, expressive piano performance evaluation differs from general audio quality assessment because the object being judged is not only audio fidelity or production quality, but also the perceived skill, fluency, and expressiveness of a performance. To support a scalable evaluation in this setting, we introduce PerformanceMOS, a neural regression model trained on rubric-augmented and human-rated piano performance data. PerformanceMOS provides a data-driven approximation of expert perceptual evaluation and complements the reconstruction-based measures when comparing expressive rendering models.

3 Methodology

YQX+ models expressive piano performance as a score-conditioned prediction problem. Given a symbolic score, the aim is to predict a compact set of note-level parameters that describe how the written notes are realized in performance. A rendering system could instead emit performance MIDI or audio directly [18, 19]; we predict parameters because benchmarking requires all model families to share one fixed target, and because parameters defined relative to score notes keep the output note-aligned to the score by construction. This entails two design choices: a representation of the musical context available from the score, and a target representation that separates expressive deviations from the underlying score content.

We first describe the score-derived features used to represent each note in context. These features extend the original YQX representation by incorporating information about pitch, harmony, rhythm, phrase structure, and score markings. We then define the expressive targets predicted by the models: beat period, velocity, timing deviation, and articulation. Finally, we describe the probabilistic modeling approaches used to compare different ways of learning the mapping from score features to expressive performance parameters.

3.1 Score and Performance Representation

3.1.1 Features

The original YQX system used seven features describing timing, articulation, and loudness, including pitch intervals, local rhythmic context, and melodic expectation features derived from Narmour's implication-realization (IR) theory. YQX+ extends this representation with five categories of score-derived features: pitch, harmony, rhythm, phrase structure, and score markings. Inspired by midihum [35], we also include long-range contextual features to capture musical information beyond the immediate neighborhood of notes. Figure 2 illustrates a representative subset of these

features. A complete list of features and descriptions is provided in the supplementary materials.

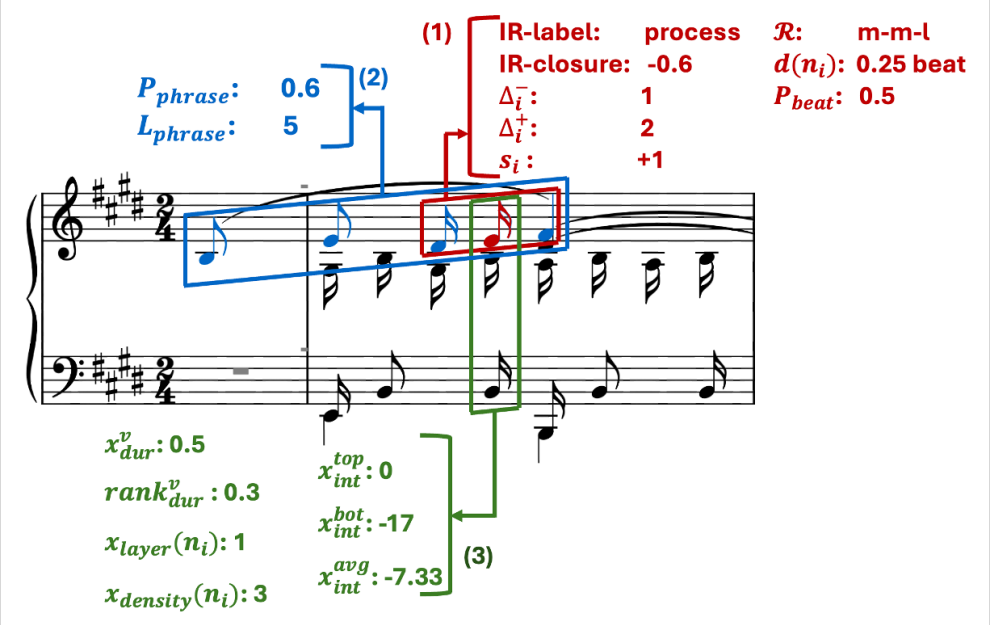


Fig. 2 An example showing some prominent features of: (1, red) the current note and the two adjacent notes; (2, blue) notes that belong to the same phrase as the current note; (3, green) notes simultaneous with the current note.

Pitch

Besides the basic pitch class and octave information, we extended the pitch features as follows:

- **Pitch interval features**: For each note n_i, forward and backward pitch intervals are computed as $\Delta_i^{+,k} = p(n_{i+k}) - p(n_i)$ and $\Delta_i^{-,k} = p(n_i) - p(n_{i-k})$, where $p(n_i)$ denotes the pitch of n_i in semitones and the index i runs over the notes of the voice containing n_i. Every interval is measured from n_i itself rather than between successive notes, and we take $k \in \{1, 2, 3\}$, giving six features per note: three forward

intervals to the first, second, and third following note of the voice, and three backward intervals to the first, second, and third preceding note. Intervals that reach beyond the start or end of the voice are set to zero.

- **Pitch interval direction**: The direction of each forward interval, $s_i^k = \text{sign}(\Delta_i^{+,k})$ for $k \in \{1, 2, 3\}$, where $+1$ is ascending, -1 is descending, and 0 represents a repeated pitch.
- **Melodic contour**: Unlike the interval features above, which are all measured from n_i, this feature describes the directions *between successive* notes. It is the sequence $(\text{sign}(p(n_{i+j+1}) - p(n_{i+j})))_{j=0}^{l-1}$ of directions over the next l steps of the voice, taken as a single categorical pattern, with $l \in \{3, 5\}$.
- **Pitch range context**: $\frac{p(n_i) - \min_{n_j \in v_{n_i}} p(n_j)}{\max_{n_j \in v_{n_i}} p(n_j) - \min_{n_j \in v_{n_i}} p(n_j)}$, the normalized pitch of n_i with respect to the pitch range of all notes n_j in its voice v_{n_i}.

Harmony

Harmony features capture interdependencies among simultaneously active musical notes, relevant for the performer's interpretation of layering and texture.

- **Harmonic density**: The *onset density* is defined as the number of notes still sounding when n_i begins, i.e., $x_{\text{density}}(n_i) = |\{n_j \mid o(n_j) \leq o(n_i) < o(n_j) + d(n_j)\}|$, where $o(n_i)$ and $d(n_i)$ are the onset time and duration of note n_i in beats. The *density ratio* normalizes this by the largest number of notes sharing a single onset anywhere in the piece.
- **Harmonic interval**: These describe pitch relations between the current note and the other notes sounding when it begins. All three are signed from n_i outwards, so that a positive value means the other note lies above n_i: the interval to the highest note n_{top} is $x_{\text{int}}^{\text{top}}(n_i) = p(n_{\text{top}}) - p(n_i)$, to the lowest note n_{bot} is $x_{\text{int}}^{\text{bot}}(n_i) = p(n_{\text{bot}}) - p(n_i)$, and to the average pitch is $x_{\text{int}}^{\text{avg}}(n_i) = \bar{p}_{\text{on}}(n_i) - p(n_i)$, where $\bar{p}_{\text{on}}(n_i)$ is the mean pitch of those notes.

Rhythm

Following the YQX system, we extract rhythm-related features that capture local patterns, beat positions, and relative durations.

- **Rhythmic context**: Local duration patterns, computed at two granularities. The *relative* pattern compares the neighbouring durations to that of n_i: each of $d(n_{i-1})$ and $d(n_{i+1})$ is labelled short (s) when its ratio to $d(n_i)$ falls below 0.75, long (l) when it exceeds 1.33, and medium (m) otherwise. The *absolute* patterns discretize the durations of n_i and its following two or four notes in the voice into short ($<$ 0.5 beats), medium ($<$ 1 beat), and long ($\geq$ 1 beat), forming sequences such as `s-s-l-m-l` that characterize the local rhythmic contour $\mathcal{R}$. Notes whose window extends past the end of the voice take a separate boundary category.
- **Beat position**: Using the measure and time signature parsed from the score, the position of n_i within its measure is $P_{\text{beat}}(n_i) = \frac{o(n_i) - o_{\text{m}}(n_i)}{L_{\text{m}}(n_i)}$, where $o_{\text{m}}(n_i)$ and $L_{\text{m}}(n_i)$ are the onset and length of the containing measure in beats. A note n_i is considered *on beat* if it lies within 0.1 beats of any metrical pulse of its measure, and *on downbeat* if it lies within 0.1 beats of the measure start. Scores without measure information fall back to a 4/4 approximation.
- **Duration context**: We compute duration ratios relative to both the average duration in the same voice and in the entire piece: $x^{v}_{\text{dur}} = \frac{d(n_i)}{\text{mean}_{n_j \in v_{n_i}} d(n_j)}$, and $x^{p}_{\text{dur}} = \frac{d(n_i)}{\text{mean}_{n_j \in p_{n_i}} d(n_j)}$, where $d(n_i)$ is the duration of note n_i. Additionally, we compute the normalized rank of n_i's duration within its voice, $\text{rank}^{v}_{\text{dur}}(n_i)$, and the piece, $\text{rank}^{p}_{\text{dur}}(n_i)$.

Phrase

Phrase boundaries are detected at rests, where the gap between the end of one note and the onset of the next exceeds 0.5 beats, and at large pitch leaps of more than 12 semitones between consecutive notes. Within each phrase, we compute:

- **Phrase position**: The normalized position of each note within its phrase $P_{\text{phrase}}(n_i)$, and the phrase length (number of notes) $L_{\text{phrase}}(n_i)$.
- **IR features**: Narmour's Implication-Realization theory [10, 11] holds that a melodic interval sets up an expectation about the next one: a small interval implies continuation in the same direction and at a similar size, while a large interval implies a change of direction and a smaller interval to follow. Each note can therefore be labelled by whether the interval that arrives fulfils or violates that implication, yielding categories such as *process* (the expected continuation), *reversal* (a change of direction), and *registral return* (the melody arrives back near where it started). Following the implementation in YQX [1], we compute: *IR-label*, derived from the implicative interval $x^{-}_{pi}(n_i)$, the realized interval $x^{+}_{pi}(n_i)$, and their product indicating registral direction; and *IR-closure*, a scalar estimating how strongly the note acts as a point of arrival rather than a continuation. These features are informative for performance because pianists tend to mark points of melodic closure by slowing down and playing more softly.

Score markings

We additionally extract performance-directive annotations from the MusicXML score. *Articulation marks* (*accent*, *staccato*, and their combination) are encoded as the indicator features $x_{\text{acc}}(n_i)$, $x_{\text{stac}}(n_i)$, and a categorical *articulation type*. *Dynamic markings* (constant markings from *pppp* to *ffff*, *sf* and *sfz*, and *cresc.*, *decresc.* and *dim.* directions) whose span covers n_i are mapped to the categorical *dynamic type*, together with a *dynamic strength* that scales the note's velocity by a factor between 0.3 (*pppp*) and 1.7 (*ffff*), and a categorical *direction* and *contour*. This matches the marking conditioning used by VirtuosoNet [2], ensuring the comparison in Section 5 is on equal footing with respect to score annotations.

3.1.2 Targets

The target representation follows Basis Mixer [13], where each performed note is described by four expressive parameters. Together with the score, these parameters provide a compact representation from which the aligned symbolic performance can be reconstructed without loss. The four note-level targets are described as follows:

- **Beat period**: The ratio of the inter-onset intervals (IOIs) between successive score onset positions in the performance and the score. This parameter is defined as follows:
  $x_{\text{bp}}(n_i) = \frac{\text{IOI}_{\text{perf}}(n_i)}{\text{IOI}_{\text{score}}(n_i)} = \frac{\hat{o}^{\text{perf}}(n_{i+1}) - \hat{o}^{\text{perf}}(n_i)}{o(n_{i+1}) - o(n_i)}$, where $\hat{o}^{perf}(n_i)$ is the actual performed onset time, in seconds, corresponding to score onset $o(n_i)$ in beats, calculated as the average onset time of all notes played at score onset position of note n_i.
- **Velocity**: $x_{\text{vel}}(n_i) = \frac{\text{vel}(n_i)}{127}$, where vel is the MIDI velocity of a played note.
- **Timing**: $x_{\text{tim}}(n_i) = \Delta_t(n_i) = \hat{o}^{\text{perf}}(n_i) - o^{\text{perf}}(n_i)$, the average onset time of all notes played at the score onset position (used in beat period), minus the performance onset time of n_i. The beat period is taken as the 'tempo grid' notion [36]; timing would then refer to the micro-deviation of each note relative to the grid.
- **Articulation**: $x_{\text{art}}(n_i) = \log_2\left(\frac{d^{\text{perf}}(n_i)}{d(n_i) \cdot x_{\text{bp}}(n_i)}\right)$, the base-2 logarithm of the ratio between the played and the expected (tempo-adjusted score) note duration. Taking the logarithm follows the partitura `encode_performance` convention and makes the scale symmetric: a note held twice as long as notated and one held half as long are equally far from zero in opposite directions, whereas on the raw ratio scale the former is 1.0 away from unity and the latter only 0.5. Since articulation is stored in these units, the D_a figures reported in Section 5, like the other deviation measures, are root mean squared errors (RMSE), but in $\log_2$ duration-ratio units rather than in seconds: $D_a = 1$ corresponds to being wrong by a factor of two in note length.

3.2 Probabilistic modeling

The selected models span a range of modeling assumptions, from the original statistical formulation of YQX to modern neural generative approaches. To establish baseline comparisons and evaluate the effectiveness of the updated feature representation, we evaluate two traditional machine learning models: the Gaussian mixture model (GMM) used in the original YQX [1], and XGBoost, following its use in midihum [35]. To go beyond these traditional statistical approaches, we also evaluate Variational Autoencoder (VAE) and Conditional Flow Matching (CFM) models within the YQX+ framework.

3.2.1 GMM

We reproduce the GMM experiments following the original YQX formulation [1] with K-Means initialization. Each target expressive parameter is modeled with a separate GMM trained on the expanded feature set. The GPU-accelerated implementation is provided by the `gmm-gpu` library [37]. We evaluate different capacities varying the number of mixture components $\{2, 4, 8, 16, 32\}$, finding that the configuration with 16 components consistently achieved the best generalization performance.

3.2.2 XGBoost

For each expressive target t, we train an independent XGBoost regressor f_t that predicts the target value $\hat{y}_t$ from the note-level feature vector $\mathbf{x} \in \mathbb{R}^d$. The model combines an ensemble of K regression trees: $\hat{y}_t(\mathbf{x}) = \sum_{k=1}^{K} f_{t,k}(\mathbf{x})$, where each tree $f_{t,k}$ incrementally corrects the residual errors of previous trees. The training is to minimize the squared error loss with a regularization term on tree complexity to avoid overfitting. The specific experiment details are reported in Section 4.

3.2.3 Variational Autoencoder

A transformer-based conditional β-VAE is implemented. The encoder takes note-wise contextual features and outputs parameters of a latent Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$, from which a latent variable z is sampled. The decoder, conditioned on both z and the original context features, reconstructs expressive targets. Both encoder and decoder are implemented as multi-layer transformer modules. The model is trained by maximizing the following β-VAE objective, which consists of a reconstruction term and a KL divergence regularization term:

$$\mathcal{L}_{\beta\text{-VAE}} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x})} \left[\log p_\theta(\mathbf{y} \mid \mathbf{z}, \mathbf{x})\right] - \beta \, D_{\mathrm{KL}} \left(q_\phi(\mathbf{z} \mid \mathbf{y}, \mathbf{x}) \parallel p(\mathbf{z})\right), \tag{1}$$

where x represents the input features, y represents the target parameters, θ denotes the parameters of the conditional likelihood $p_\theta(\mathbf{y}|\mathbf{z}, \mathbf{x})$, ϕ denotes the parameters of the approximate posterior $q_\phi(\mathbf{z}|\mathbf{y}, \mathbf{x})$, $p(\mathbf{z}) = \mathcal{N}(0, I)$ is the latent prior, and β is the hyperparameter controlling the weight of the KL divergence term.

3.2.4 Conditional Flow Matching

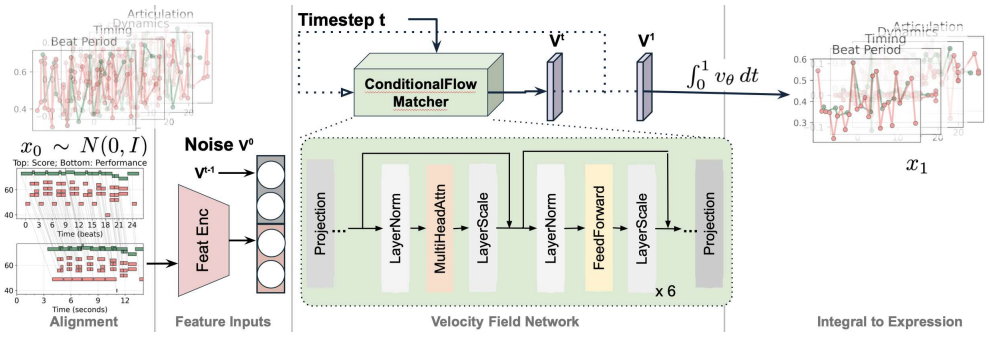


Fig. 3 Overview of the framework for expressive performance generation conditioned on score-level features. The conditional flow matcher takes a symbolic score and random noise $x_0 \sim \mathcal{N}(0, I)$ as input and predicts the velocity field ν_t iteratively throughout the trajectory. The output of the velocity field network is mapped to the final expressive performance space, capturing timing, dynamics, articulation, and beat period variations.

We aim to model a continuous vector field $v_\theta(x_t, t, f)$ over expressive performance parameters x, where $t \in [0, 1]$ is a time parameter interpolating between a noise prior and the real data distribution, and f represents the musical features. The training objective is to approximate the conditional vector field that transports samples from an initial Gaussian noise $x_0 \sim N(0, I)$ to the target expressive parameter x_1, by following trajectories that approximate either linear interpolations or optimal transport (OT) paths in the data space.

As illustrated in Figure 3, we adopt the CFM objective [38], which trains a neural network to approximate a target velocity field u_t. Our CFM model v_θ is parameterized by a transformer architecture [39]. Given a source sample x_0 and a target sample x_1 drawn jointly from the data distribution, we construct intermediate points x_t by interpolating between x_0 and x_1 with added Gaussian noise $x_t \sim \mathcal{N}\big(t\, x_1 + (1 - t)\, x_0,\ \sigma^2 I\big)$ where $t \sim \mathcal{U}[0, 1]$. The network is trained to match the ground-truth conditional velocity field $u_t\left(x_t \mid x_1\right) = \frac{x_1 - x_t}{1 - t}$ by minimizing the mean squared error:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{x_0, x_1, t}\left[||v_\theta(x_t, t, f) - u_t||_2^2\right]. \tag{2}$$

We consider both the linear and the optimal transport (OT) CFM objectives. Regarding OT CFM where the pairings (x_0, x_1) are computed via an OT plan within each mini-batch, the joint distribution $\pi(x_0, x_1)$ is approximated by solving:

$$\pi^* = \arg \min_{\pi \in \Pi(\mu_0, \mu_1)} \langle C, \pi \rangle, \tag{3}$$

where C is the pairwise cost matrix. Samples $(x_0, x_1) \sim \pi^*$ are used to compute the interpolants x_t and the velocity learning.

4 Experiment setup

4.1 Datasets

For training, we used the score-performance aligned datasets ASAP [40] and ATEPP [41], which contain MusicXML scores and aligned performance MIDI for 1815 pieces from the Western classical repertoire. The Vienna4x22 dataset [42], which contains 22 performances of four excerpts, served as the test dataset for all experiments. We evaluated three feature configurations, corresponding to short, long, and full contexts, to assess the impact of different levels of musical context on model performance.

- **Short context**: This configuration includes only local features, directly derived from the target note and its immediate neighbors. It represents the smallest temporal window and focuses on note-level details without broader musical context.
- **Long context**: This configuration extends the local representation with phrase-level structure and long-term dependencies, including multistep melodic intervals, interval directions over two- or three-step windows, rhythmic patterns across multiple notes, and statistical measures over voice- and piece-level pitch and duration distributions.
- **Full context**: This configuration incorporates all available features, including the extended context from the midihum feature set and technical indicators, providing a comprehensive and multi-scale representation of musical structure.

4.2 Experiment details

For the XGBoost model, we followed the midihum implementation, using a learning rate of 0.05, maximum depth of 7, 1700 estimators, subsampling of 0.9, and column sampling of 0.6 to enhance generalization. We also conducted experiments with higher-capacity configurations, increasing the maximum depth to 9 and the number

of estimators to 2500. Both the β-VAE and CFM architectures were trained with approximately 1.6M parameters, 6 Transformer layers, 8 attention heads, a hidden dimension of 128, and, for the β-VAE, a latent dimension of 64. Models were trained for up to 1000 epochs using the Adam optimizer with an initial learning rate of 10^{-4}, a cosine learning-rate scheduler, and Xavier uniform weight initialization. All training was conducted on a high-performance computing cluster using a Tesla V100 GPU.

5 Evaluation

We evaluate YQX+ models using three complementary approaches. First, we compare predicted expressive parameters against aligned human performances using objective error and correlation metrics. Second, we use PerformanceMOS as an automatic perceptual proxy to estimate perceived performance quality from symbolic piano performances. Third, we conduct a subjective listening study to assess whether the model outputs are perceived as natural, expressive, and musically convincing by human listeners. Together, these evaluations allow us to examine both parameter-level accuracy and perceptual quality.

5.1 Objective metrics

We evaluated the expressive performance predictions using the objective metrics computed against the Vienna4x22 dataset. We calculated the RMSE between predicted and ground-truth parameters, capturing the magnitude of deviation across each parameter. We also measured Pearson correlation coefficients for tempo and velocity curves, C_{temp} and C_{vel}, at the onset level to evaluate the structural similarity of expressive trajectories. The results shown in Table 2 were aggregated across 22 performances per piece, and 95% confidence intervals were computed for all metrics.

The larger XGBoost configuration (m9, n2500) performed on-par with our best performing experiment of β-VAE, particularly for beat period, timing, and velocity,

Table 2 Feature ablation study across different models and context windows. For XGBoost, parameters m7 and m9 represent maximum depths of 7 and 9, respectively, while n1700 and n2500 represent 1700 and 2500 estimators, respectively. Each model is evaluated with three feature configurations: Short context (S), Long context (L), and Full context (F). D_b, D_t, D_v, and D_a represent the RMSE of beat period, timing, velocity, and articulation, respectively (D_a in $\log_2$ duration-ratio units), and C_{temp} and C_{vel} the onset-level Pearson correlation of the tempo and velocity curves. Arrows indicate the preferred direction of each metric. GMM, Gaussian mixture model; VAE, variational autoencoder; CFM, conditional flow matching; MOS, the predicted PerformanceMOS score.

Model	**Feats**		**Objective Metrics**						**MOS ↑**
	params		$D_b \downarrow$	$D_t \downarrow$	$D_v \downarrow$	$D_a \downarrow$	$C_{\textbf{temp}} \uparrow$	$C_{\textbf{vel}} \uparrow$	
DExter	-	-	0.57±0.32	0.10±0.06	**0.23**±0.02	1.40±0.25	**0.29**±0.17	0.06±0.06	59.0
VirtuosoNet	-	-	0.54±0.24	0.12±0.11	0.24±0.15	1.14±0.24	0.21±0.12	**0.08**±0.24	64.4
GMM	$n_c = 16$	S	0.64±0.54	0.05±0.01	0.28±0.04	1.62±0.16	0.01±0.03	0.03±0.08	64.4
	$n_c = 16$	L	0.66±0.54	0.04±0.02	0.28±0.04	1.66±0.16	0.04±0.11	-0.01±0.06	64.3
XGBoost	$m7, n1700$	S	0.53±0.62	0.05±0.01	0.25±0.05	1.69±0.46	0.22±0.07	-0.06±0.18	61.2
	$m7, n1700$	L	0.70±0.25	0.06±0.02	0.26±0.11	4.65±0.32	-0.09±0.13	**0.08**±0.14	62.2
	$m7, n1700$	F	0.54±0.48	0.11±0.01	0.24±0.04	1.24±0.34	-0.03±0.03	0.02±0.09	64.6
	$m9, n2500$	S	0.60±0.65	0.06±0.01	0.26±0.10	2.05±0.42	0.10±0.10	0.01±0.12	62.4
	$m9, n2500$	L	0.51±0.62	0.04±0.02	0.24±0.06	1.61±0.45	0.15±0.08	0.05±0.18	63.5
	$m9, n2500$	F	0.51±0.57	0.04±0.02	0.24±0.05	1.60±0.47	0.03±0.04	0.03±0.10	62.5
β-VAE	$\beta = 4$	S	0.51±0.61	**0.03**±0.02	0.24±0.06	1.02±0.26	0.03±0.07	0.02±0.07	60.2
	$\beta = 4$	L	**0.50**±0.61	**0.03**±0.02	0.24±0.04	1.38±0.37	0.23±0.07	0.05±0.15	62.4
	$\beta = 1$	F	0.53±0.63	**0.03**±0.02	**0.23**±0.06	1.14±0.31	0.13±0.07	0.02±0.11	62.0
	$\beta = 4$	F	0.53±0.63	**0.03**±0.02	0.25±0.12	**0.95**±0.20	-0.08±0.08	0.02±0.11	64.5
	$\beta = 10$	F	**0.50**±0.60	**0.03**±0.02	0.25±0.12	1.51±0.42	0.22±0.09	0.02±0.10	**68.9**
CFM	linear	S	0.61±0.56	0.05±0.01	0.27±0.03	1.68±0.19	0.02±0.08	0.03±0.10	52.9
	linear	L	5.27±0.55	1.20±0.08	0.30±0.04	3.03±0.18	0.05±0.05	-0.05±0.09	50.3
	linear	F	0.89±0.62	0.07±0.01	0.35±0.15	1.64±0.14	0.01±0.05	0.05±0.07	60.2
	OT	S	0.59±0.56	0.04±0.02	0.29±0.04	2.20±0.24	0.02±0.08	0.03±0.04	52.0
	OT	L	1.05±0.17	0.16±0.04	0.40±0.09	3.14±0.13	-0.10±0.06	0.02±0.07	51.7
	OT	F	0.87±0.32	0.04±0.02	0.25±0.03	1.99±0.28	-0.001±0.05	0.02±0.06	59.8

although its articulation errors remained higher than the best β-VAE configuration. Among the β-VAE models, the full-context setting with $\beta = 4$ achieved the lowest articulation error ($D_a = 0.95$), while the $\beta = 10$ setting achieved the lowest beat-period error among the β-VAE variants and the highest PerformanceMOS score reported in Table 2. In contrast, the CFM models showed substantial variation across trajectory objectives and feature configurations, with some settings producing markedly higher errors.

Compared with the baselines DExter and VirtuosoNet, several YQX+ configurations obtained lower timing deviation (D_t) and articulation deviation (D_a), although the improvements were not consistent across all metrics. Tempo and velocity correlations remained low across the YQX+ models, with occasional negative correlations. This suggests that while the models can predict local expressive parameters with relatively low error, they do not yet consistently capture the larger-scale temporal and dynamic trajectories learned by sequential models such as DExter and VirtuosoNet.

Across model families, the difference between short and long contexts was small in most cases. Full-context settings provided modest improvements in correlation metrics for XGBoost ($C_{vel} = 0.08$) and β-VAE ($C_{\text{temp}} = 0.22$). However, deviation metrics (D_b, D_t, D_v, D_a) were largely similar across context lengths, with no consistent trend favoring a specific context size.

5.2 PerformanceMOS model

5.2.1 Dataset and Augmentation

Accurately predicting perceptual ratings of expressive piano performance remains challenging in music education and music information retrieval [24], partly because of the difficulty of specifying objective assessment criteria [22, 28]. To obtain training data covering a wide range of performance quality, we constructed a rule-based

mistake-synthesis pipeline that generates corrupted performances of varying severity. The rubric was informed by the Expert-Novice dataset [22, 26] and by ABRSM grading criteria.

Algorithm 1: PerformanceMOS Data Augmentation with Rating Penalties

Input: Dataset $\mathcal{D}$ of performances with maximum rating $R_{\max}$ and score-performance alignment.

$P_{\text{uniform}}(a, b)$: random value in $[a, b]$
$\mathcal{U}(0, 1)$: uniform probability over $[0, 1]$
foreach $P \in \mathcal{D}$ **do**
  $r \leftarrow R_{\max}$;
  if $\mathcal{U} < 0.5$ **then**
    Sample intensity $\iota \sim \mathcal{U}(0.1, 1)$; corrupt $n = \min(\lfloor \iota\,|P| \rfloor, 10)$ wrong notes; $r \leftarrow r + n \cdot P_{\text{uniform}}(-3, -1)$;
  if $\mathcal{U} < 0.5$ **then**
    Sample $\iota \sim \mathcal{U}(0.1, 1)$; corrupt $n = \min(\lfloor \iota\,|P| \rfloor, 12)$ timing values; $r \leftarrow r + n \cdot P_{\text{uniform}}(-2, -0.5)$;
  if $\mathcal{U} < 0.5$ **then**
    Sample $\iota \sim \mathcal{U}(0.1, 1)$; corrupt $n = \min(\lfloor \iota\,|P| \rfloor, 15)$ velocities; $r \leftarrow r + n \cdot P_{\text{uniform}}(-1.5, -0.3)$;
  if $\mathcal{U} < 0.2$ **then**
    Remove a voice; $r \leftarrow r + P_{\text{uniform}}(-60, -30)$;
  if $\mathcal{U} < 0.2$ **then**
    Remove a passage; $r \leftarrow r + P_{\text{uniform}}(-50, -20)$;
  $r \leftarrow \max(1, r)$; store (P, r);

For each clean performance in ASAP [40] and ATEPP [41], five corruption types are sampled independently as Bernoulli events, as shown in Algorithm 1. These corruptions include incorrect notes, timing errors, velocity errors, missing voices, and missing passages. Multiple corruptions may be applied to the same sample, and each applied corruption subtracts a penalty from the maximum rating $R_{\max}$ of the source dataset, where $R_{\max} = 80$ for ASAP and $R_{\max} = 99$ for ATEPP. To prevent long pieces from receiving unbounded penalties, the number of corrupted notes is capped for the small-error categories: 10 for wrong notes, 12 for timing errors, and 15 for velocity errors.

The constants in Algorithm 1 are design choices and encode three assumptions. First, since r starts at $R_{\max}$ and only decreases, $R_{\max}$ is the rating an uncorrupted performance receives, and therefore expresses the assumed quality ceiling of each source corpus. Second, the per-note penalty ranges order the three small-error types by how salient they are to a listener, with wrong notes ($[-3, -1]$) penalised more heavily than timing errors ($[-2, -0.5]$) and these in turn more heavily than velocity errors ($[-1.5, -0.3]$); the structural corruptions are penalised an order of magnitude more, on the assumption that a missing voice or passage is categorically worse than any accumulation of local errors. Third, the caps are set in the inverse order of the per-note penalties so that the three categories contribute comparable worst-case totals ($10 \times 3 = 30$, $12 \times 2 = 24$, $15 \times 1.5 = 22.5$), which keeps any one category from dominating the synthetic rating. Together these produce a continuous distribution of synthetic quality scores suitable for regression training, as illustrated in Figure 4.

In addition to the rubric-augmented data, we trained PerformanceMOS using the Transkun-transcribed [43] YCU-PPE-III dataset [25]. Each performance in this dataset was rated independently by three human judges, and we used the average of the three scores as the training label for MOS prediction.

5.2.2 Model Architecture and Training

Our PerformanceMOS model builds upon Aria [44], a self-supervised transformer architecture pre-trained on large-scale MIDI performance data for music understanding tasks. We adapt its fine-tuning classification variant by replacing the classification head with a regression head to predict continuous MOS scores on a 0-100 scale. We report the best result from fine-tuning the model end-to-end on our data, following Aria's fine-tuning protocol with a learning rate of 10^{-5} (linear scheduler), batch size of 12, and training for 10 epochs.

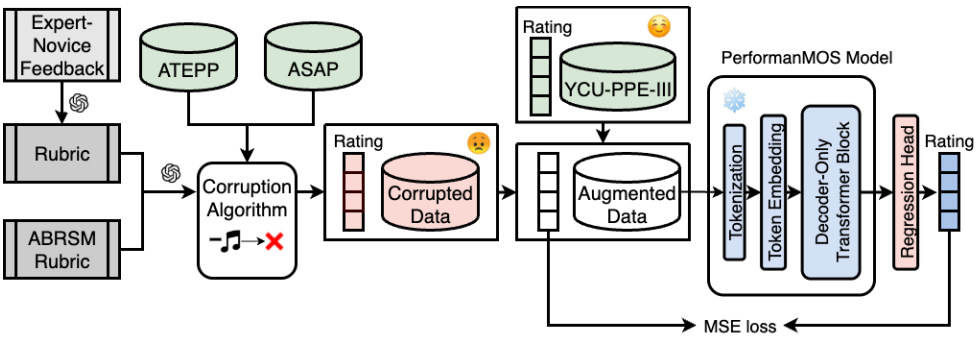


Fig. 4 Design overview for the corruption algorithm and PerformanceMOS pipeline.

5.2.3 PerformanceMOS Results

The PerformanceMOS predictions are shown in the final column of Table 2. The highest predicted MOS was obtained by the β-VAE model with $\beta = 10$ and full context (68.9). The best XGBoost configuration also achieved a competitive predicted MOS score (64.6), slightly above DExter (59.0) and comparable to VirtuosoNet (64.4). Several other YQX+ configurations, including GMM and β-VAE variants, obtained predicted MOS scores in the same range as VirtuosoNet, while the CFM models consistently received lower predicted MOS scores.

The PerformanceMOS results therefore suggest that the strongest β-VAE and XGBoost configurations are competitive with existing expressive rendering baselines in terms of predicted perceptual quality. However, the relationship between objective errors and predicted MOS is not perfectly direct: some models achieve low parameter-level errors without obtaining the highest MOS score. This supports the use of PerformanceMOS as a complementary perceptual proxy rather than a replacement for objective metrics or human listening tests.

5.3 Subjective Listening Experiment

We conducted a MUSHRA-style listening evaluation on the four Vienna4x22 excerpts (Frédéric Chopin: Étude in E major Op. 10, No. 3, bars 1–21; Frédéric Chopin: Ballade

in F major Op. 38, 1–45; Franz Schubert: Deutscher Tanz in f minor D.783 No. 15; Wolfgang Amadeus Mozart: Sonata in A major K.331, first movement, bars 1–18) [42]. Forty listeners, including 22 with more than five years of classical music engagement (listening and playing experience), rated six rendered versions of each excerpt on a 0 to 100 scale across four dimensions: naturalness, rhythmic freedom, expressiveness, and performance quality. The music score was rendered using the same soundfont, with its music markings approximately interpreted by the MuseScore[1] system, the rendered audio was provided as a reference. This resulted in $40 \times 4 \times 6 \times 4 = 3{,}840$ ratings. Individual listeners agreed only weakly with one another (single-listener $\mathrm{ICC}(2,1) = 0.057$; mean pairwise Pearson $\bar{r} = 0.09$; Krippendorff's $\alpha = 0.041$), which is expected when expressive nuance is judged by a heterogeneous listener panel rather than by calibrated experts [30]. Averaging over the panel is nevertheless reliable: $\mathrm{ICC}(2, k{=}40) = 0.707$ is estimated from the same between-listener and between-stimulus variance components and expresses the reliability of the 40-listener mean rather than of any single rating. Subsequent analyses therefore use ratings aggregated over listeners. By comparison, the three trained YCU-PPE-III judges used to train PerformanceMOS showed higher panel-level reliability ($\mathrm{ICC}(2, k{=}3) = 0.846$, single-judge leave-one-out $r = 0.69$–0.77), consistent with rubric-based expert assessment.

Table 3 reports the mean ratings for each system with 95% bootstrap confidence intervals. The β-VAE model achieved the highest mean rating (62.7), followed closely by VirtuosoNet (61.1) and XGBoost (60.1). DExter (56.4), CFM (51.7), and GMM (50.4) received lower mean ratings. Paired Wilcoxon tests with Holm correction over all 15 system pairs placed the top three systems in a statistical tie ($p_{\mathrm{Holm}} \geq 0.09$; Cliff's $\delta \leq 0.10$, negligible), while all nine comparisons between the top-three and bottom-three systems were significant ($p_{\mathrm{Holm}} < 0.01$). These results indicate that the learned probabilistic YQX+ models, particularly XGBoost and β-VAE, reach

[1] https://musescore.org

perceptual parity with the strongest prior baseline, VirtuosoNet, and substantially outperform DExter, CFM, and GMM in the listening test.

Rhythmic freedom is not desirable in itself, and how much of it is appropriate depends on the repertoire, so we verified that this dimension does not drive the aggregate. Excluding it and averaging the remaining three dimensions leaves the ordering unchanged (β-VAE 62.5, VirtuosoNet 61.6, XGBoost 60.2, DExter 56.5, CFM 51.1, GMM 50.1).

Table 3 Listening test: mean rating across all pieces and dimensions, 95% bootstrap CI ($n = 160$).

System	Mean	95% CI
β-VAE	62.7	[59.4, 66.0]
VirtuosoNet	61.1	[57.7, 64.5]
XGBoost	60.1	[57.0, 63.3]
DExter	56.4	[53.1, 59.8]
CFM	51.7	[48.1, 55.3]
GMM	50.4	[46.7, 54.0]

The ranking varied substantially by piece, as shown in Figure 5. VirtuosoNet achieved the highest ratings on both Chopin excerpts (Op. 10 No. 3: 66.0; Op. 38: 64.4), XGBoost achieved the highest rating on Mozart K. 331 (66.1, compared with 65.2 for β-VAE), and β-VAE achieved the highest rating on Schubert D. 783 No. 15 (62.3). The four perceptual dimensions were moderately correlated ($r = 0.62$–0.69) and largely preserved the same ordering, except that XGBoost tied with VirtuosoNet for rhythmic freedom. Experienced listeners gave ratings that were uniformly 2–7 points higher across systems, without changing the comparative ordering, supporting the pooled analysis.

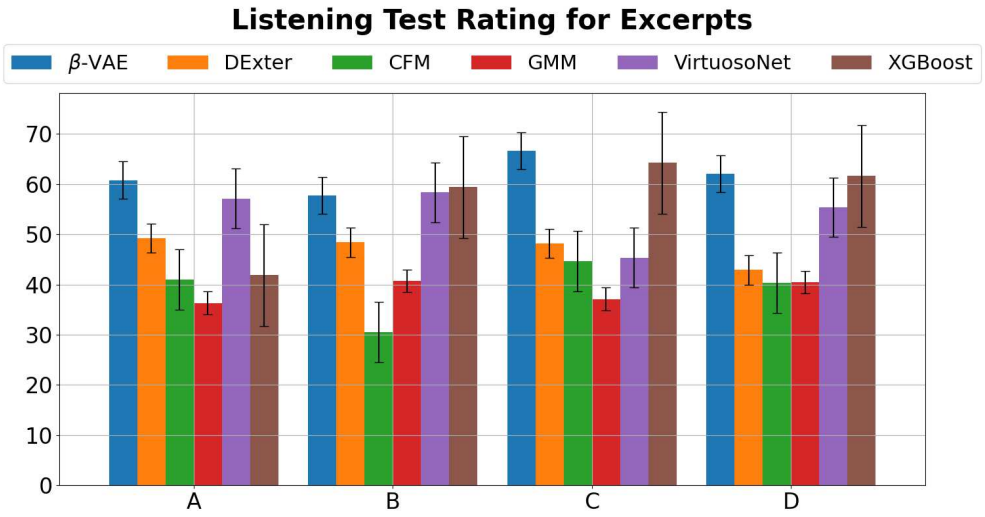


Fig. 5 Listening test rating for excerpts: (A) Chopin Op.10 No.3, (B) Chopin Op.38, No.2, (C) Mozart K.331, and (D) Schubert D.783, No.15.

5.3.1 Comparison with PerformanceMOS

On the YCU-PPE-III test split ($n = 251$), PerformanceMOS predictions correlated strongly with the three-judge mean (Pearson $r = 0.87$, Spearman $\rho = 0.90$, MAE = 3.79, RMSE = 6.43). Per-judge correlations ranged from $r = 0.69$ to 0.85, indicating that PerformanceMOS agrees closely with the expert consensus and exceeds the single-judge leave-one-out consistency range reported above ($r = 0.69$–0.77).

On the Vienna4x22 listening test, evaluated at the piece × system level ($n = 24$), PerformanceMOS showed a weaker but positive association with listener ratings (Pearson $r = 0.28$, Spearman $\rho = 0.39$, Kendall $\tau = 0.33$, MAE = 15.0). The model correctly identified the top-rated system for three of the four pieces: both Chopin excerpts and Schubert, with Mozart K. 331 within one rank. Per-dimension correlations were strongest for naturalness ($r = 0.36$) and performance quality ($r = 0.28$), with rhythmic freedom and expressiveness at approximately $r = 0.19$. This pattern is consistent with PerformanceMOS being trained on YCU-style examiner ratings, which emphasize overall cleanness and naturalness.

The lower correlation on Vienna4x22 is expected, since these stimuli are synthesized model outputs rather than recorded student performances, and the listener panel is more heterogeneous than the expert ratings used for training. Overall, PerformanceMOS acts as an expert-aligned predictor within its training domain and as a useful trend indicator for synthesized expressive renderings, but it should be interpreted as a complement to, rather than a replacement for, human listening evaluation.

6 Discussion and Conclusion

YQX+ recasts expressive performance rendering as a probabilistic prediction problem over a compact and interpretable target space: beat period, velocity, timing, and articulation. By combining this target representation with shared datasets, feature configurations, objective metrics, predicted MOS scores, and listening-test evaluation, YQX+ supports systematic comparison across different model families. On Vienna4x22, XGBoost and β-VAE reach perceptual parity with VirtuosoNet, with the β-VAE $\beta = 10$ configuration achieving the highest predicted MOS. Neither requires recurrence or iterative sampling, in contrast to the hierarchical recurrent architecture of VirtuosoNet and the diffusion sampling loop of DExter, so parity is reached at a lower inference cost. The combination of a 40-participant listening study and the YCU-PPE-III expert panel also provides a useful basis for evaluating automatic perceptual predictors: the listening study gives a crowd-rated benchmark on synthesised renderings, while YCU-PPE-III provides a higher-reliability expert-rated dataset against which PerformanceMOS achieves a strong correlation with expert consensus.

PerformanceMOS provides a fast and reproducible perceptual proxy that complements objective metrics and can help identify cases where full listening tests are most needed. The four note-level targets used in YQX+ are recoverable from any aligned score-performance corpus, making the framework extensible to broader repertoire,

additional targets such as pedal control, and transformer-based controllable rendering systems such as ScorePerformer retrained under the same target representation. Future work could also connect YQX+ to neural audio synthesis pipelines, enabling downstream comparison between symbolic performance quality and audio-domain perceptual quality.

References

[1] G. Widmer, S. Flossmann, M. Grachten, YQX Plays Chopin. AI Magazine **30**(3), 35–48 (2009). https://doi.org/10.1609/aimag.v30i3.2249

[2] D. Jeong, T. Kwon, Y. Kim, K. Lee, J. Nam, VirtuosoNet: A Hierarchical RNN-based System for Modeling Expressive Piano Performance. In Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR) p. 908–915 (2019)

[3] C.E. Cancino-Chacón, M. Grachten, W. Goebl, G. Widmer, Computational Models of Expressive Music Performance: A Comprehensive and Critical Review. Frontiers in Digital Humanities **5**(October), 1–23 (2018). https://doi.org/10.3389/fdigh.2018.00025

[4] M. Hashida, T. Nakra, H. Katayose, T. Murao, K. Hirata, K. Suzuki, T. Kitahara, et al., Rencon: Performance Rendering Contest for Automated Music Systems. In Proceedings of the 10th International Conference on Music Perception and Cognition (ICMPC) (2008)

[5] H. Zhang, S. Chowdhury, C.E. Cancino-Chacón, J. Liang, S. Dixon, G. Widmer, DExter: Learning and Controlling Performance Expression with Diffusion Models. Applied Sciences **14**(15) (2024). https://doi.org/10.3390/app14156543

[6] H. Zhang, A. Maezawa, S. Dixon. RenderBox: Expressive Performance Rendering with Text Control (2025). Preprint at https://arxiv.org/abs/2502.07711

[7] G. Widmer, S. Dixon, W. Goebl, E. Pampalk, A. Tobudic, In Search of the Horowitz Factor. AI Magazine **24**(3), 111–130 (2003)

[8] G. Widmer, Discovering Simple Rules in Complex Data: A Meta-Learning Algorithm and Some Surprising Musical Discoveries. Artificial Intelligence **146**(2), 129–148 (2003). https://doi.org/10.1016/S0004-3702(03)00016-X

[9] R. Bresin, A. Friberg, J. Sundberg, Director Musices : The KTH Performance Rules System. Special Interest Group on Music and Computer(SIGMUS) - 46 pp. 43–48 (2002)

[10] E. Narmour, The Analysis and Cognition of Basic Melodic Structures: The Implication-Realization Model. University of Chicago Press (1990)

[11] M.T. Pearce, G.A. Wiggins, Expectation in melody: The influence of context and learning. Music Perception: An Interdisciplinary Journal **23**(5), 377–405 (2006). https://doi.org/10.1525/mp.2006.23.5.377

[12] T.H. Kim, S. Fukayama, T. Nishimoto, S. Sagayama, Polyhymnia: An Automatic Piano Performance System with Statistical Modeling of Polyphonic Expression and Musical Symbol Interpretation. In Proceedings of the International Conference on New Interfaces for Musical Expression pp. 96–99 (2011)

[13] C.E. Cancino-Chacón, Computational Modeling of Expressive Music Performance with Linear and Non-linear Basis Function Models. Ph.D. thesis, Johannes Kepler University Linz (2018)

[14] I. Borovik, D. Gavrilev, V. Viro, SyMuPe: Affective and Controllable Symbolic Music Performance. In Proceedings of the 33rd ACM International Conference on Multimedia (MM'25) pp. 10699–10708 (2025). https://doi.org/10.1145/3746027.3755871

[15] A. Maezawa, K. Yamamoto, T. Fujishima, Rendering Music Performance with Interpretation Variations Using Conditional Variational RNN. In Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR) pp. 855–861 (2019)

[16] S. Rhyu, S. Kim, K. Lee, Sketching the Expression: Flexible Rendering of Expressive Piano Performance with Self-Supervised Learning. In Proceedings of 23rd the International Society on Music Information Retrieval (ISMIR) pp. 178–185 (2022)

[17] I. Borovik, V. Viro, ScorePerformer : Expressive Piano Performance Rendering with Fine-grained Control. In Proceedings of the 24th International Society on Music Information Retrieval (ISMIR) pp. 588–596 (2023)

[18] S. Oore, I. Simon, S. Dieleman, D. Eck, K. Simonyan, This Time with Feeling: Learning Expressive Musical Performance. Neural Computing and Applications **32**(4), 955–967 (2018). https://doi.org/10.1007/s00521-018-3758-9

[19] H. Zhang, S. Dixon, Disentangling the Horowitz Factor: Learning Content and Style From Expressive Piano Performance. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2023)

[20] S.D. Peter, C.E. Cancino-chacón, G. Widmer, Sounding Out Reconstruction Error-Based Evaluation of Generative Models of Expressive Performance. In Proceedings of the 10th International Conference on Digital Libraries for Musicology

(DLfM) pp. 58–66 (2023). https://doi.org/10.1145/3625135.3625141

[21] A. Lerch, C. Arthur, A. Pati, S. Gururani, An Interdisciplinary Review of Music Performance Analysis. Transactions of the International Society for Music Information Retrieval (TISMIR) **3**(1), 221–245 (2020). https://doi.org/10.5334/tismir.53

[22] Y. Jiang, Expert and Novice Evaluations of Piano Performances : Criteria for Computer-Aided Feedback. In Proceedings of the 24th International Society on Music Information Retrieval (ISMIR) p. 367–374 (2023)

[23] J. Park, J. Kim, J.M. Park, A. Choi, W.S. Li, J. Park, S.w. Hwang, Piano Performance Evaluation Dataset with Multilevel Perceptual Features. Scientific Reports **14**(23002) (2024). https://doi.org/10.1038/s41598-024-73810-0

[24] H. Zhang, V. Cheung, H. Nishioka, S. Dixon, S. Furuya, How Does the Teacher Rate? Observations from the NeuroPiano Dataset. Late-Breaking and Demo Session of the International Society on Music Information Retrieval (ISMIR LBD) (2024)

[25] W. Wang, J. Pan, H. Yi, Z. Song, M. Li, Audio-Based Piano Performance Evaluation for Beginners with Convolutional Neural Network and Attention Mechanism. IEEE/ACM Transactions on Audio Speech and Language Processing **29**, 1119–1133 (2021)

[26] H. Zhang, V. Cheung, H. Nishioka, S. Dixon, S. Furuya, LLaQo: Towards a Query-Based Coach in Expressive Music Performance Assessment. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 1–5 (2025)

[27] A. Morsi, K. Tatsumi, A. Maezawa, T. Fujishima, X. Serra, Sounds Out of Pläce? Score-Independent Detection of Conspicuous Mistakes in Piano Performances.

In Proceeding of the 24th International Society on Music Information Retrieval (ISMIR) pp. 352–358 (2023). https://doi.org/10.5281/zenodo.10265297

[28] A. Morsi, H. Zhang, A. Maezawa, S. Dixon, X. Serra, Simulating Piano Performance Mistakes for Music Learning. In Proceedings of the 21st Sound and Music Computing Conference (SMC) pp. 179–186 (2024)

[29] S. Chang, S. Dixon, E. Benetos, RUMAA: Repeat-Aware Unified Music Audio Analysis for Score-Performance Alignment, Transcription, and Mistake Detection. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (2025)

[30] A. Tjandra, Y.C. Wu, B. Guo, J. Hoffman, B. Ellis, A. Vyas, B. Shi, S. Chen, M. Le, N. Zacharov, C. Wood, A. Lee, W.N. Hsu, Meta Audiobox Aesthetics: Unified Automatic Quality Assessment for Speech, Music, and Sound. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) pp. 1–17 (2025)

[31] J. Yao, G. Ma, H. Xue, H. Chen, C. Hao, Y. Jiang, H. Liu, R. Yuan, J. Xu, W. Xue, H. Liu, L. Xie. SongEval: A Benchmark Dataset for Song Aesthetics Evaluation (2025). Preprint at http://arxiv.org/abs/2505.10793

[32] Y. Yuan, H. Liu, J. Liang, X. Liu, M.D. Plumbley, W. Wang, Leveraging Pre-Trained AudioLDM for Sound Generation: A Benchmark Study. In 2023 31st European Signal Processing Conference (EUSIPCO) pp. 765–769 (2023)

[33] W.C. Huang, S.W. Fu, E. Cooper, R.E. Zezario, T. Toda, H.M. Wang, J. Yamagishi, Y. Tsao, The Voicemos Challenge 2024: Beyond Speech Quality Prediction. IEEE Spoken Language Technology Workshop (SLT) pp. 803–810 (2024)

[34] H. Zhang, J. Liang, H. Phan, W. Wang, E. Benetos, From Aesthetics to Human Preferences: Comparative Perspectives of Evaluating Text-to-Music Systems. IEEE 35th International Workshop on Machine Learning for Signal Processing (MLSP) pp. 1–6 (2025)

[35] E. Waldemarsson. midihum: A Set of Scripts for Converting MIDI to Humdrum. https://github.com/erwald/midihum (2021). Accessed: 2025-07-29

[36] J. Gillick, J. Yang, C.E. Cella, D. Bamman, Drumroll Please: Modeling Multi-Scale Rhythmic Gestures with Flexible Grids. Transactions of the International Society for Music Information Retrieval (TISMIR) **4**(1), 156–166 (2021). https://doi.org/10.5334/tismir.98

[37] gmm-gpu: Gaussian mixture models on gpu. https://pypi.org/project/gmm-gpu/ (2025). Accessed: 2025-07-23

[38] Y. Lipman, R.T.Q. Chen, H. Ben-Hamu, M. Nickel, M. Le, Flow Matching for Generative Modeling. In Proceedings of the 11th International Conference on Learning Representations (ICLR) (2023)

[39] O. Tal, A. Ziv, I. Gat, F. Kreuk, Y. Adi, Joint Audio and Symbolic Conditioning for Temporally Controlled Text-to-Music Generation. In Proceedings of the 25t International Society on Music Information Retrieval (ISMIR) pp. 264–271 (2024)

[40] F. Foscarin, A. Mcleod, P. Rigaux, F. Jacquemard, M. Sakai, ASAP : A Dataset of Aligned Scores and Performances for Piano Transcription. In Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR) pp. 534–541 (2020)

[41] H. Zhang, J. Tang, S. Rafee, S. Dixon, G. Fazekas, ATEPP: A Dataset of Automatically Transcribed Expressive Piano Performance. In Proceedings of the 23nd

International Society for Music Information Retrieval Conference (ISMIR) pp. 446–453 (2022)

[42] W. Goebl. The Vienna 4x22 Piano Corpus. http://dx.doi.org/10.21939/4X22 (1999). https://doi.org/10.21939/4X22. Accessed: 2017-01-31

[43] Y. Yan, Z. Duan, Scoring Time Intervals Using Non-Hierarchical Transformer for Automatic Piano Transcription. In Proceedings of the 25th International Society on Music Information Retrieval (ISMIR) p. 973–980 (2024)

[44] L. Bradshaw, H. Fan, A. Spangher, S. Biderman, S. Colton, Scaling self-supervised representation learning for symbolic piano performance. In Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR) p. 451–459 (2025)

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- supplementary.zip