Descriptor-Aware Latent Granular Mosaicing for Real-Time Audio Re-Synthesis

Alfred Pichard **Stefan Lattner**

Sony Computer Science Laboratories

Abstract

We introduce descriptor-aware latent granular mosaicing, a method for real-time audio reconstruction based on grain selection in the latent space of a neural audio autoencoder (Pasini et al. (2024)). Prior work shows that granular synthesis in neural codec latent spaces enables corpus-based transformation by matching grains between source and target signals (Tokui and Baker (2025)), but relies solely on latent similarity and offers limited additional perceptual control. We address this by augmenting latent grain retrieval with learned perceptual descriptors, including loudness, transientness, brightness, and spectral flatness. Grain selection is formulated as a multi-factor distance combining latent proximity and descriptor similarity, enabling continuous control while preserving source structure. The method is implemented as Latent Mosaicer, a Max for Live device for interactive DAW use. Evaluation on Slakh-based codebooks and sources (Manilow et al. (2019)) measures descriptor tracking and indicates that explicit descriptor goals can steer retrieval in the rendered output.

1 INTRODUCTION

Audio mosaicing and corpus-based concatenative synthesis transform a source signal by reconstructing it from units drawn from a target corpus while preserving selected aspects of timing and texture (Truax (1988); Roads (2001); Schwarz (2007); Thibault (2024)). In most such systems, retrieval is driven by hand-crafted signal descriptors, whose main advantage is interpretability: users can steer transformations through perceptually meaningful dimensions such as brightness, loudness, noisiness, or pitch. However, descriptor-based distances remain only partial proxies of auditory similarity and may fail to capture the higher-order timbral and temporal structure.

Recent neural audio codecs and autoencoding models provide compact latent representations that encode salient perceptual regularities beyond manually designed features (Zeghidour et al. (2021); Défossez et al. (2022); Kumar et al. (2023); Pasini et al. (2024)). This has motivated latent-space approaches to audio transformation (Caillon and Esling (2022)), including latent granular resynthesis, in which a codebook of latent grains is queried to reconstruct a source signal by nearest-neighbor matching in latent space (Bitton et al. (2020); Tokui and Baker (2025)). Such methods often produce perceptually coherent results and preserve broad structural and timbral relationships without requiring explicit descriptor engineering.

However, this strength comes with a limitation: latent similarity alone offers little direct control over specific perceptual attributes. Although it supports plausible matching, it does not readily expose independent control over dimensions such as brightness, transientness, or spectral noisiness. This is a significant limitation in interactive musical contexts, where the usefulness of a transformation system depends not only on perceptual quality but also on the availability of stable and interpretable control parameters.

In this work, we propose descriptor-aware latent granular mosaicing, a retrieval framework that combines latent similarity with learned perceptual descriptors within a unified grain selection criterion. Rather than replacing latent matching, the method augments it with descriptor-aware objectives so that retrieved grains remain close to the source in latent space while also following target perceptual tendencies. In the present implementation, these targets include loudness, transientness, brightness, and spectral flatness. The aim is to preserve the perceptual co-

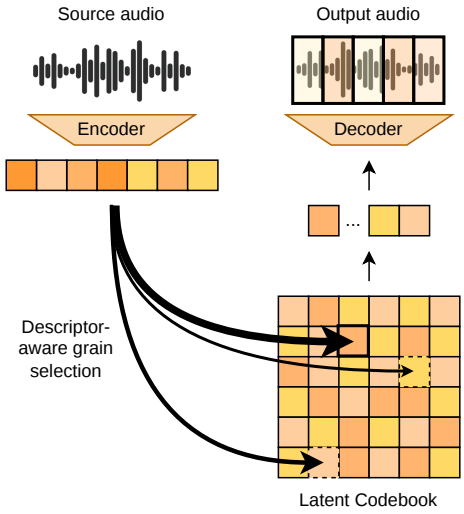

Figure 1: Latent Mosaicer grain selection is a combination of latent distance and descriptor weights. Top-K latent closest grains are ordered with their descriptor similarity (arrows thickness) and selected with a temperature parameter to be decoded in real-time.

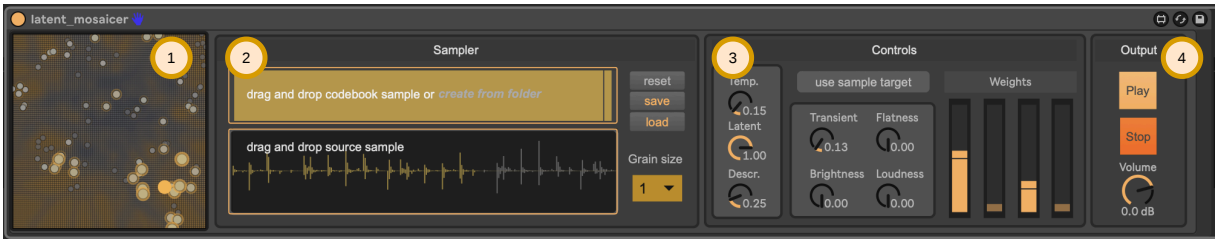

Figure 2: The *Latent Mosaicer* Max for Live device. 1-Codebook and accessible grains visualization; each grain is scaled relative to the probability it has to be selected under the current control targets. 2-Sampler section; allows the user to load, save and reset the codebook, define the source sample and select the grain size. 3- Controls section; left part includes temperature, latent weight, descriptors weight. Middle part is for target values and allows the user to select either targets from the source sample or to select custom values. Right part is used to control each descriptor strength. From left to right : Transient, Flatness, Brightness, Loudness. 4-Output utility panel; with play, stop and volume control.

herence of latent retrieval while restoring explicit control dimensions more typical of descriptor-based concatenative systems.

We further demonstrate the method in a real-time Max for Live implementation designed for the DAW-based interaction[1]. This makes it possible to evaluate the approach not only as an offline retrieval procedure but also as a practical instrument for controlled sound transformation. To study its behavior, we use Slakh-based codebooks (Manilow et al. (2019)) and evaluate both codebook reachability and control-following accuracy. The results show that descriptor-aware retrieval improves controllability over purely latent matching.

The contributions of this work are threefold: (1) a descriptor-aware formulation of latent grain retrieval that extends latent mosaicing with explicit perceptual control; (2) a real-time implementation for interactive use in a DAW environment; (3) an evaluation protocol assessing both descriptor-space reachability and control accuracy.

2 SYSTEM OVERVIEW

The system builds on *music2latent* as the analysis–synthesis backbone : music2latent is a neural audio compression model that encodes audio into a compact sequence of 64-dimensional latent vectors and decodes these representations back into audio (Pasini et al. (2024)). A reference corpus is processed into a codebook of latent grains, each consisting of one or more consecutive latent frames, depending on the selected grain size. This formulation follows recent latent granular resynthesis approaches, where the audio is reconstructed by matching source grains to its nearest neighbors in a latent codebook (Tokui and Baker (2025)). In such systems, retrieval is typically based only on latent similarity. While effective in preserving global structure and timbre consistency, this strategy provides limited control over perceptual attributes.

We extend this paradigm by introducing descriptor-aware retrieval. For each incoming source grain, the system computes a similarity score against all codebook grains. In baseline mode, this score is given by the cosine distance in the latent space. In the proposed formulation, it is augmented with distances in a learned descriptor space which captures loudness, transientness, brightness, and spectral flatness. Grain selection is thus expressed as a weighted multi-objective criterion combining latent proximity and descriptor targets. The selected latent grain is decoded back to audio, producing an output signal entirely constructed from the target codebook. This preserves the structural advantages of latent mosaicing while opening the way for explicit perceptual control.

Our method is implemented in *Latent Mosaicer* (Fig. 2), a Max for Live device developed as a practical interface for descriptor-aware latent granular resynthesis taking inspiration from existing instruments (Schwarz et al. (2006); Thibault (2024)). Rather than exposing the system only as an offline analysis script, we present the retrieval process as a compact but fully operational instrument inside Ableton Live. Its design follows the logic of the underlying pipeline: users first define a target corpus, then load a source signal to be transformed, and finally shape the retrieval process through a small set of controls that set temporal granularity, stochasticity, and descriptor targets. In this sense, the interface is not only a wrapper, but also an explicit operationalization of the proposed framework for interactive and creative musical use (Schwarz et al. (2006)).

Codebook construction can be performed from a single audio file or from a folder of files. Each file is encoded with *music2latent*, segmented into latent grains at the selected grain size, and stored together with predicted descriptor values for loudness, transientness, brightness, and spectral flatness. The device also supports bringing additional material to an existing codebook, resetting it, and saving or loading prepared codebooks, which avoids recomputing descriptors during later sessions. A separate source signal is then loaded and encoded in the same latent space. During playback, the source determines the temporal sequence of queries, while the codebook defines the available replacement material. Hence, the output is constructed from the target corpus, but its temporal organization follows the source according to the retrieval settings.

The main controls expose the parameters of the retrieval model. The *grain size* sets the number of consecutive latent frames that are used as a matching unit. The *temperature* controls whether the selection is nearly deterministic or more exploratory. The *latent weight* and descriptor weights set the balance between matching the source in latent space and following perceptual targets. Individual weights are provided for transientness, flatness, brightness, and loudness. Descriptor targets can be used in two modes : in source-goal mode, the target descriptor values are estimated from the source grain itself, so retrieval follows the

[1]Device and demos are available for download at https://github.com/AlfredPichard/latent_mosaicer

evolving perceptual profile of the input. In user-goal mode, the source descriptor values are replaced by manually selected targets, allowing the user to push the output toward a fixed region of the codebook descriptor space. Mode selection is operated through a button on the control part of the interface.

To support an interactive understanding of the process, the device also includes a compact visualization of the codebook organization. A two-dimensional projection of the latent codebook is shown in the interface with dynamic highlighting related to the current retrieval state. Although this view is not intended as an analytical figure in itself, it provides immediate feedback on how the system navigates the corpus during playback and helps users connect audible changes to movements through the latent space. In combination with the source waveform display and transport controls, this makes the retrieval process legible enough for performance and experimentation rather than remaining a black-box resynthesis engine.

Taken together, these elements make *Latent Mosaicer* a usable exploration tool: it allows rapid comparison of corpora, controlled manipulation of perceptual targets, reproducible recall of prepared codebooks, and real-time audition of the consequences of descriptor-aware retrieval. This makes it suitable both for formal evaluation and for exploratory compositional use, where users may move continuously between source-preserving mosaicing and more strongly transformed, descriptor-driven resynthesis.

3 METHOD

3.1 Latent Grain Codebook

An input audio signal is encoded into a latent sequence $z = (z_1, ..., z_T)$, where each frame $z_t \in \mathbb{R}^d$ is produced by a neural audio encoder – with $d = 64$ in *music2latent* (Pasini et al. (2024)). Given a grain size g, the sequence is partitioned into non-overlapping latent grains with a fixed hop size, which provides a consistent segmentation of the audio for codebook construction and retrieval.

$$G_i = (z_{i*g}, ..., z_{(i+1)*g-1}) \tag{1}$$

Applying this procedure to a reference corpus yields a codebook $\mathcal{C} = \{C_k\}$ of latent grains, while applying it to an input signal produces source grains to be reconstructed. This formulation follows the original latent granular resynthesis approach (Tokui and Baker (2025)), where reconstruction is performed via nearest-neighbor matching in latent space.

3.2 Grain Pooling and Descriptor Prediction

In order to keep the descriptor prediction lightweight and the input dimensionality fixed, we perform a pooling into a fixed-dimensional representation by concatenating the grain-wise latent mean and standard deviation. Hence, each grain $G_i \in \mathbb{R}^{g \times d}$ is pooled along its temporal axis: we compute the mean and standard deviation across the g latent frames independently, yielding :

$$\psi(G_i) = [\mu(G_i), \sigma(G_i)] \tag{2}$$

This representation captures the distribution of the grain in latent space, providing a more stable basis for descriptor estimation. The perceptual descriptors are then predicted as follows:

$$\phi_{\mathcal{D}}(G_i, g) = f_{\mathcal{D}}(\psi(G_i), g) \tag{3}$$

where $\phi_{\mathcal{D}}$ denotes the estimate of the descriptor $\mathcal{D}$, and each $f_{\mathcal{D}}$ is a descriptor-specific learned model. Our implementation includes loudness, transientness, brightness, and spectral flatness (Peeters (2004)). Further research could explore less explicit descriptors and their relation to perceptual control.

3.3 Descriptor-aware Retrieval

Given a source grain G and a codebook $\mathcal{C} = \{C_k\}$, the task is to select the best match grain $C_k \in \mathcal{C}$ for resynthesis. In the original latent granular formulation, retrieval is typically performed by nearest-neighbor search in latent space :

$$C^* = \arg\min_{C_k \in \mathcal{C}} D_{\text{lat}}(G, C_k) \tag{4}$$

where D_{latent} measures the similarity with a cosine distance between grains. We extend this formulation by introducing descriptor-aware retrieval. Each grain is associated with descriptor estimates $\phi_{\mathcal{D}}(.)$ and the selection is based on a combined distance :

$$\begin{aligned} D(G, C_k) &= w_{\text{lat}} D_{\text{lat}}(G, C_k) \\ &+ \sum_{\mathcal{D}} w_{\mathcal{D}} D_{\mathcal{D}}(\phi_{\mathcal{D}}(G), \phi_{\mathcal{D}}(C_k)) \end{aligned} \tag{5}$$

Here, $D_{\mathcal{D}}$ denotes a distance in the descriptor space, and each term is normalized using statistics computed over the codebook to ensure comparable scales. The weights w_{lat} and $w_{\mathcal{D}}$ define a trade-off between source similarity and perceptual alignment. To enable explicit control, descriptor values for the source grain can be replaced by user-defined targets. Denoting the resulting descriptor vector by $\mathbf{d}_{\text{target}}$, the retrieval becomes :

$$\begin{aligned} C^* &= \arg\min_{C_k \in \mathcal{C}} w_{\text{lat}} D_{\text{lat}}(G, C_k) \\ &+ \sum_{\mathcal{D}} w_{\mathcal{D}} D_{\mathcal{D}}(\mathbf{d}_{\text{target}}, \phi_{\mathcal{D}}(C_k)) \end{aligned} \tag{6}$$

This formulation can be interpreted as a scalarized multi-objective nearest-neighbor search, where latent proximity preserves structural coherence while descriptor terms steer perceptual characteristics. For example, taken a given source grain, one codebook grain may be closer in latent space, while another better matches its transient. Adjusting these weights therefore controls the trade-off between these two objectives rather than optimizing both simultaneously. Optionally, selection can be relaxed using a softmax distribution with temperature τ that provides a probabilistic generalization of nearest-neighbor retrieval.

$$p(C_k \mid G) \propto \exp[(-\frac{1}{\tau} D(G, C_k)] \tag{7}$$

3.4 Real-Time Generation

The method is implemented as a custom Max external integrating TorchScript models with a JavaScript-based interface, enabling deployment as a Max for Live device for real-time use. The device exposes parameters for codebook construction and descriptor-aware retrieval, providing direct control over grain selection behavior within a DAW environment. Moreover, the interface includes a two-dimensional UMAP projection of the codebook grains, offering a reduced representation of the latent space (McInnes et al. (2018)). This visualization allows users to inspect the distribution of grains and observe how the

retrieval trajectories evolve under different settings of grain size, temperature, and descriptor weighting. This design supports both exploratory interaction and controlled transformation by linking high-dimensional retrieval processes to interpretable controls. While the primary contribution lies in the descriptor-aware latent retrieval formulation, its integration into an interactive system demonstrates its applicability to real-time audio effects and computer music practice.

4 EXPERIMENTAL SETUP AND TRAINING DETAILS

4.1 Training Details

The descriptor predictors were trained on the training split using a mixed Slakh/NSynth loader, with Slakh sampled with probability 0.7, and evaluated on the validation split (Manilow et al. (2019); Engel et al. (2017)). The regression targets for these heads are waveform-derived descriptors, computed directly from grain-aligned audio segments. For each segment, descriptors are extracted from an STFT with Hann window, $n_{\text{fft}} = 1024$, and hop size 256. Loudness is computed from the mean log RMS magnitude, transientness from positive spectral flux, brightness from the log spectral centroid, and flatness from the ratio between geometric and arithmetic spectral magnitude means.

Each descriptor head is a grain-size-conditioned MLP operating on pooled latent grains. For a grain $G_i \in \mathbb{R}^{g\times 64}$, we compute the mean and standard deviation across the g latent frames independently for each latent dimension and concatenate them into a 128-dimensional vector. The network uses hidden dimensions $(128, 64)$; a learned grain-size embedding is added after the first linear layer, followed by ReLU activations and a descriptor-specific scalar output. The scalar heads are trained with an L_1 regression loss using Adam with learning rate 10^{-3}.

4.2 Experimental Setup

The following end-to-end reachability and controllability experiments (section 5) use only the Slakh test split. In this evaluation, a *corpus stem* denotes an isolated Slakh stem used to build a retrieval codebook, while a *source stem* denotes an isolated stem used as the input trajectory to be reconstructed. For each grain size, we sample three coarse stem groups : drums, bass, and lead, chosen to cover complementary temporal and timbral regimes while keeping musically meaningful instrument classes. Each group contributes four corpus stems and six source stems. Pairing every corpus stem with every source stem within the same group gives $4 \times 6 \times 3 = 72$ corpus-source pairs per grain size. Codebooks are built from 20s corpus excerpts, while source excerpts are 6s long. With the *music2latent* hop used here, these 6s source excerpts correspond to 65, 32, 21, 16, 13, 9 and 7 source grains for $g = \{1, 2, 3, 4, 5, 7, 9\}$ respectively.

For the offline evaluation, retrieval is performed over the full codebook built from each corpus excerpt. In the final controllability evaluation, descriptors are recomputed on the decoded output audio using the same waveform-domain procedure described above. The real-time Max external uses a precomputed list of candidates for efficiency, with default $K = 50$, and selects from this set using the same weighted latent-descriptor distance and temperature parameter.

5 EVALUATION

We evaluate the proposed system along three complementary axes: (i) the reliability of the grain-level descriptor predictors, (ii) the intrinsic reachability of target grains from the latent codebook, and (iii) the extent to which descriptor-aware retrieval yields controllable descriptor trajectories in the rendered audio. This decomposition is important because end-to-end control can only succeed when two conditions are jointly met: the descriptors used for steering must be estimated reliably and the codebook must contain locally reachable grains that realize the requested descriptor change.

5.1 Descriptor-Prediction Evaluation

Before evaluating control, we first verify that the pooled latent grain representation preserves enough information to predict the perceptual descriptors used for steering. The descriptor heads are evaluated on the validation split with 319,488 sampled grains, using grain-size-conditioned MLPs trained to regress waveform-derived teacher descriptors from pooled latent statistics. Table 1 reports the global mean absolute error and Spearman rank correlation for the four scalar descriptors used later in the retrieval objective.

In general, loudness and transientness are the most reliable descriptor heads, with global Spearman correlations of 0.774 and 0.801, respectively. Brightness and flatness require a more cautious interpretation. Their global correlations are lower, and their per-grain-size behavior is not monotonic. Brightness shows useful rank agreement for several grain sizes but degrades sharply at some resolutions, while flatness is globally less stable. This instability should not be interpreted as a smooth function of g: changing the grain size also changes the non-overlapping segmentation, the number of validation grains, and the learned grain size conditioning.

The unstable cases are better interpreted as failures of rank ordering rather than uniformly large absolute errors. For example, brightness at some larger grain sizes keeps an MAE similar to other resolutions, while its Spearman correlation drops, suggesting that the predictor preserves a coarse scale but not the ordering of individual grains. Flatness is particularly sensitive because it is a nonlinear spectral ratio and may not be well represented by temporal mean and variance pooling alone. These results support the use of learned descriptor heads for steering, but with limits: loudness and transientness provide the most stable control signals, whereas brightness and flatness should be understood as weaker, scale-dependent ranking cues.

Table 1: Global descriptor-prediction results on the validation split.

Descriptor	MAE	Spearman ρ
Loudness	16.727	0.774
Transient	1.805	0.801
Brightness	1.197	0.364
Flatness	0.348	0.416

5.2 Reachability

Controllability should not be evaluated independently of the corpus itself: if a target grain has no sufficiently close neighbor in the codebook, then any failure in descriptor tracking may just reflect corpus mismatch rather than a lim-

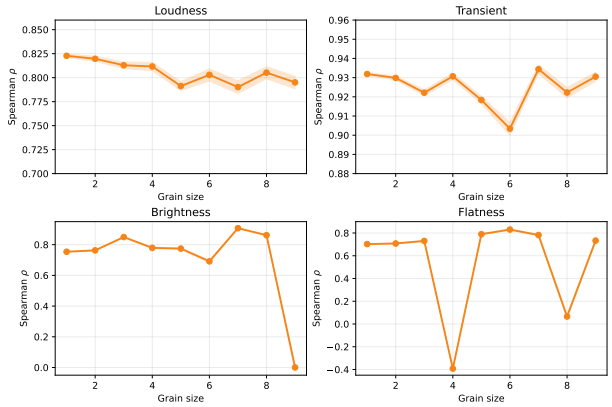

Figure 3: Spearman correlation between predicted and waveform-derived descriptor values as a function of grain size. Loudness and transientness are comparatively stable, whereas brightness and especially flatness show irregular, grain-size-dependent behavior.

itation of the steering mechanism. We therefore characterize local codebook coverage through the nearest-neighbor latent distance $d_{\min}$, defined as the minimum cosine distance between a target grain and all codebook grains. This quantity provides a direct, threshold-free measure of how closely the target is represented in the available latent vocabulary.

The primary result is that the empirical distribution of $d_{\min}$ remains concentrated at low values across grain sizes. Globally, the mean nearest-neighbor distance is 0.120, with median 0.095, $q_{05} = 0.019$, and $q_{95} = 0.278$. Table 2 shows that this behavior remains stable across temporal resolutions: the mean $d_{\min}$ increases only moderately with grain size, from 0.105 at $g = 1$ to 0.132 at $g = 9$. This indicates that, for most target grains, the codebook contains a close latent surrogate rather than only distant approximations. For interpretability, we additionally report a binary reachability ratio obtained by thresholding $d_{\min}$ at $\tau_g = 0.47$, used as a secondary summary statistic. This threshold was fixed before the reported evaluation from preliminary in-domain Slakh calibration runs as a conservative upper-tail cutoff for nearest-neighbor latent distances, and was then kept constant across grain sizes. Under this binarization, reachability remains high across all tested grain sizes, varying only between 0.950 and 0.968, with a mean ratio of 0.957.

Because nearest-neighbor distances are generally low and thresholded reachability remains high, subsequent steering measurements can be interpreted primarily as measurements of retrieval and control effectiveness rather than as artifacts of a sparse or badly mismatched latent codebook. Hence, the reachability analysis shows that the Slakh test set is locally dense enough for the controllability results to be meaningful.

Table 2: Latent reachability by grain size (1-5-9).

g	Reachability	Mean $d_{\min}$	q_{95}
1	0.959	0.105	0.300
5	0.956	0.126	0.283
9	0.958	0.132	0.257

5.3 End-to-End Descriptor Control

This final evaluation measures whether descriptor-aware retrieval produces the intended descriptor trajectories in the rendered waveform. Descriptors are computed on the decoded audio rather than on latent predictions, so this stage evaluates the full chain from goal specification to grain retrieval to audio rendering. We use a latent-only baseline, where descriptor weights are set to zero and retrieval is driven only by latent proximity, and a user-target condition, where one descriptor at a time follows a scripted trajectory, while the remaining descriptors stay at neutral values. The user trajectories include `ramp_up`, `ramp_down`, `step_up`, and `step_down`. The evaluation does not involve human participants. User goals refer to manually specified target trajectories that simulate user control inputs. We report directional accuracy, response slope, Pearson correlation, normalized MAE, and t_{90} – defined as the time required for the output descriptor to reach 90% of its final change after the control step. Directional accuracy is included only as a local sign-change diagnostic: because it compares successive descriptor differences, it can penalize small jitter or plateaus even when the overall trajectory correlation and tracking error improve.

The main result is that explicit user goals substantially outperform the latent-only baseline. Averaged over all descriptors and grain sizes, user goals reach a Pearson correlation of 0.499 with the requested trajectory, compared to 0.232 for latent-only retrieval. In practical terms, this means that when the user asks the system to progressively increase or decrease a descriptor, the measured descriptor of the resulting audio tends to follow that requested direction more closely than when the system relies only on latent similarity. The same pattern appears in tracking error: user goals reduce the normalized MAE to 0.320, versus 0.442 for the baseline. The user-goal condition also produces a strong positive follow gap, with $\Delta_{\text{follow}} = 0.484$, confirming that the output follows the imposed user trajectory much closer than the descriptor evolution preserved by latent matching alone.

The descriptor-wise behavior is summarized in Table 3 and Fig. 4. In Table 3, the row "User goals" is the aggregate result over all four controlled descriptors, all trajectories and all grain sizes. The following rows break down the same user-goal condition by the descriptor being controlled. Transientness is the easiest descriptor to steer ($r = 0.534$, MAE $= 0.291$), followed by brightness ($r = 0.509$) and loudness ($r = 0.501$). Flatness remains controllable but is the weakest of the four ($r = 0.453$). Step responses remain overall short, with the mean t_{90} ranging from 102 ms for transientness to 217 ms for flatness.

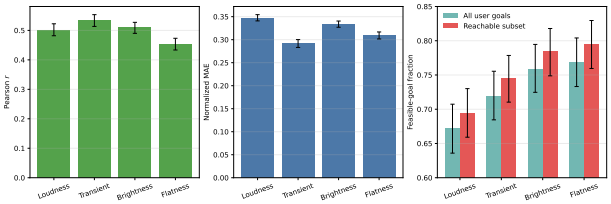

Figure 4: Descriptor-wise user-goal performance. Left: goal-following correlation. Middle: normalized tracking error. Right: fraction of user goals that fall inside a locally reachable descriptor interval, computed for all goals and for the subset with reachable latent neighbors.

As shown in Fig. 5, the effect of grain size is comparatively modest. Under user goals, Pearson correlation stays between 0.447 and 0.532, with the best average result at $g = 9$. This suggests that descriptor steering is robust across temporal resolutions, although the trade-off between responsiveness and stability remains musically relevant: larger grains appear to stabilize descriptor tracking, while shorter grains retain finer temporal detail.

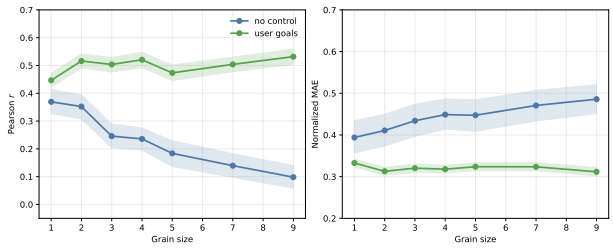

Figure 5: Controllability mean values over all descriptors, across grain sizes. User goals consistently improve goal-following correlation and reduce normalized tracking error relative to the latent-only baseline.

Finally, the end-to-end results should be read jointly with the reachability analysis. Since target grains are usually well represented in latent space, the remaining gap to perfect descriptor tracking is unlikely to be explained by corpus mismatches alone. It is therefore better interpreted as a limitation of the retrieval process and of the descriptor structure available within the local latent neighborhood. In this sense, the proposed system provides effective descriptor-level steering but not unconstrained control.

Table 3: End-to-end control metrics aggregated over all grain sizes. "User goals" reports the aggregate across all controlled descriptors, while descriptor-named rows report the same user-goal condition restricted to one controlled descriptor.

–	Dir. acc.	Slope	Pearson r	MAE	t_{90} (ms)
No control	0.530	0.144	0.232	0.442	–
User goals	0.476	0.394	0.499	0.320	156
Loudness	0.463	0.446	0.501	0.348	130
Transient	0.492	0.484	0.534	0.291	102
Brightness	0.484	0.358	0.509	0.334	173
Flatness	0.467	0.286	0.453	0.309	217

6 CONCLUSION

We presented descriptor-aware latent granular mosaicing, a retrieval framework that extends latent granular resynthesis with explicit perceptual control. By combining latent proximity with learned grain-level descriptors, the method preserves the structural coherence of latent matching while exposing interpretable control dimensions for loudness, transientness, brightness, and spectral flatness. We further implemented this approach as *Latent Mosaicer*, a real-time Max for Live device, showing that the method can function not only as an offline retrieval procedure but also as a practical instrument for interactive sound transformation.

The evaluation supports this contribution along three complementary axes. First, the descriptor heads retain sufficient alignment with waveform-domain measurements to support descriptor-aware retrieval, especially for loudness and transientness. Second, the reachability analysis shows that target grains are usually well represented in the latent codebook, allowing end-to-end control results to be interpreted as properties of the retrieval mechanism rather than artifacts of corpus mismatch. Third, explicit user goals consistently improve descriptor tracking over a latent-only baseline, demonstrating that the proposed formulation provides effective and measurable perceptual steering.

At the same time, the results clarify the limits of the approach. Control is not unconstrained: its accuracy depends on the descriptor structure available within the local latent neighborhood, and some descriptors remain less stable than others. Hence, future work should focus both on stronger descriptor models and on interaction strategies that help users navigate these local constraints musically, for example through adaptive multi-scale retrieval or corpus-aware control mappings. The choice of descriptors employed in this paper may also be subject to discussion. Future research could investigate a broader class of control strategies, including approaches that may be less transparent or interpretable than ours. More broadly, this work suggests that latent-space audio mosaicing can become a more legible and playable creative instrument when perceptually meaningful controls are embedded directly into the retrieval process.

This work was developed in the context of deep learning applied to audio, with particular attention to small and accessible interactive tools for real-time musical practice. More broadly, it reflects our interest in systems that remain usable without task-specific retraining at performance time, so that zero-shot transformation can support exploratory and immediate forms of interaction. We hope that the present device contributes both as a research instrument and as a modest practical step toward more legible and musician-centered creative AI systems.

AUTHOR DECLARATIONS

AI-assisted tools were used in the preparation of this paper to support the implementation of selected code components, including parts of the Max external for *music2latent*, and to refine the wording and clarity of the manuscript. AI was not used to design, analyze, or interpret any part of this work. All proposed implementations and text were carefully reviewed and validated by the authors.

REFERENCES

Bitton, A., Esling, P., and Harada, T. (2020). Neural granular sound synthesis.

Caillon, A. and Esling, P. (2022). Rave: A variational autoencoder for fast and high-quality neural audio synthesis. In *International Conference on Learning Representations (ICLR)*.

Défossez, A., Copet, J., Synnaeve, G., and Adi, Y. (2022). High fidelity neural audio compression.

Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D., and Simonyan, K. (2017). Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1068–1077. JMLR.org.

Kumar, R., Seetharaman, P., Luebs, A., Kumar, I., and Kumar, K. (2023). High-fidelity audio compression with improved rvqgan.

Manilow, E., Wichern, G., Seetharaman, P., and Roux, J. L. (2019). Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity.

McInnes, L., Healy, J., Saul, N., and Großberger, L. (2018). Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.

Pasini, M., Lattner, S., and Fazekas, G. (2024). Music2latent: Consistency autoencoders for latent audio compression.

Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the cuidado project. Technical report, IRCAM.

Roads, C. (2001). *Microsound*. MIT Press.

Schwarz, D. (2007). Corpus-Based Concatenative Synthesis. *IEEE Signal Processing Magazine*, 24(2):92–104. cote interne IRCAM: Schwarz07a.

Schwarz, D., Beller, G., Verbrugghe, B., and Britton, S. (2006). Real-time corpus-based concatenative synthesis with catart. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Montreal, Canada.

Thibault, D. (2024). Mosaïque - Concatenative Synthesis Instrument for the Practicing Musicians. In *Proceedings of the Conference on AI Music Creativity (AIMC)*. https://aimc2024.pubpub.org/pub/buh7kcah.

Tokui, N. and Baker, T. (2025). Latent granular resynthesis using neural audio codecs.

Truax, B. (1988). Real-time granular synthesis with a digital signal processor. *Computer Music Journal*, 12(2):14–26.

Zeghidour, N., Luebs, A., Omran, A., Skoglund, J., and Tagliasacchi, M. (2021). Soundstream: An end-to-end neural audio codec.