

PERCEPTUALLY MOTIVATED ALIGNMENT AND INTERPOLATION OF PITCH-ALIGNED TIME-FREQUENCY REPRESENTATIONS

Shahan Nercessian, Jeff Sontag, and Alejandro Koretzky

Splice
New York, USA

{shahan.nercessian|jeff.sontag|alejandro.koretzky}@splice.com

ABSTRACT

We propose a framework for the alignment and interpolation of pitch-aligned time-frequency representations. Building on the tonal interval vector, we introduce a series of extensions that reformulate it as an invertible operator, culminating in a new feature extractor that embeds perceptual consonance priors within a pitch-aligned representation. Accordingly, we develop methods for aligning and interpolating between musical structures. For alignment, we cast the problem as a permutation search under perceptually weighted distances, enabling robust matching without explicit key detection, which is inherently ambiguous in music information retrieval. For interpolation, we employ optimal transport to generate musically meaningful transitions between pitch distributions, proposing a circular formulation over the Circle of Fourths/Fifths to respect harmonic structure. Finally, we learn a low-dimensional geometric representation of scale structure aimed to factorize scale color and density. Experimental results demonstrate the effectiveness and musicality of the proposed methods. Together, our contributions provide a principled, purely signal processing approach to modeling pitch structure in audio and symbolic music signals.

1. INTRODUCTION

A longstanding objective in music information retrieval and audio signal processing is the development of pitch-aligned time-frequency representations (PTFRs) describing pitch content in audio and symbolic music signals. Representations ranging from constant-Q transforms and their derived chroma and pitch class profile (PCP) features [1], to transcription-based piano rolls and pitch activation maps [2], provide structured views of pitched material at varying levels of abstraction. PTFRs underpin a wide range of tasks, including chord recognition and key estimation [3].

Beyond analysis, these representations invite operations such as alignment, interpolation, and general transformation of pitched material, enabling re-harmonization and cross-context comparison. However, existing methods are limited in scope, or may rely on standard Euclidean formulations [4] that do not fully reflect perceptual consonance or harmonic proximity, yielding outputs that are mathematically consistent but musically implausible. The perceptually motivated tonal interval vector (TIV) [5] offer a compelling alternative, but has been used primarily for analysis purposes [6]. These limitations become apparent when manipulating

pitch structure across musical contexts. As musical phrases derive their identity through their relationship to harmonic context, instrumentation, and surrounding material, it begs the problem of how to preserve their structure while maintaining perceptual coherence. Adapting a chord progression to a new melody, modulating between tonalities, or aligning independently generated layers all require development of a “toolkit” that is musically relevant. These tasks require mathematical tools that enable such adaptation while preserving perceptually meaningful musical relationships.

Recent advances in neural audio synthesis have further amplified the importance of structured manipulation of pitch content. Systems such as MusicGen [7] and diffusion-based approaches [8] demonstrate high-quality musical generation, but typically offer limited tools to control harmonic structure. In practice, this makes it difficult to perform operations such as re-harmonization, cross-key adaptation, and coherent alignment between independently generated musical layers. Building on these advances, there is growing interest to bridge the gap between high-level musical descriptions and low-level audio manipulations. This, in turn, motivates the development of methods that enable more fine-grained and perceptually meaningful manipulation of pitch content, both as standalone tools for analysis and composition, as well as interfaces for enriching generative models. Such methods can unlock the potential of generative systems to perform structured transformations of pitch content beyond the realm of standard signal processing.

In this work, we introduce perceptually motivated methods for aligning and interpolating PTFRs. Expanding on the TIV formulation, we develop a series of extensions reformulating it as an invertible operator, enabling its application to arbitrary-dimensional pitch representations while preserving its correlation to perceptual consonance. This gives rise to a means for alignment by determining optimal permutations under perceptually weighted distances. For interpolation, we build on optimal transport (OT) techniques [9], proposing a circular extension for chroma vectors along the Circle of Fourths/Fifths (CoF) [10]. Finally, we introduce a geometric scale representation providing an interpretable low-dimensional organization of pitch distributions that factorizes into root, density, and color components. These contributions establish a principled signal processing approach to modeling and manipulating pitch structure in both audio and symbolic music, with implications for conditional generative music systems.

This paper is organized as follows: Section 2 overviews the TIV; Section 3 introduces our proposed TIV extensions; Section 4 applies these extensions to PTFR alignment; Section 5 explores musical interpolation; Section 6 develops a geometric scale representation; Section 7 reports experimental results; Section 8 draws conclusions and outlines future work.

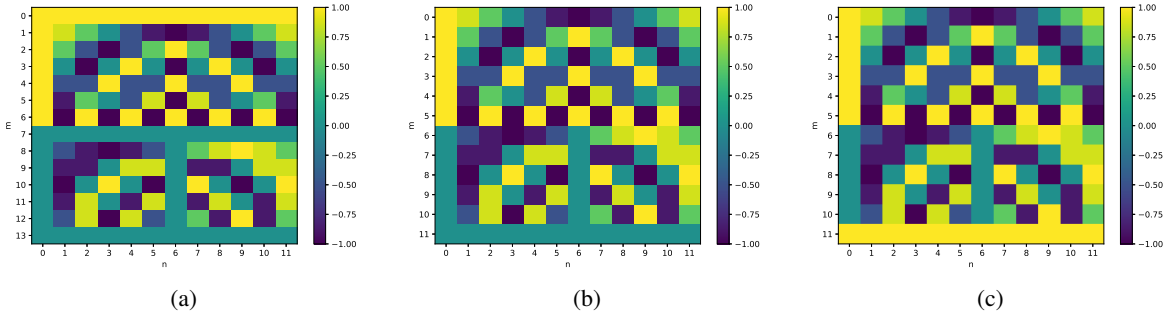


Figure 1: (a) F_{real} , (b) F implicitly used by equation (1) in [5, 6], (c) proposed invertible F .

2. TONAL INTERVAL VECTORS

Given a chroma vector $\mathbf{c} \in \mathbb{R}^N$ (with $N = 12$), its corresponding TIV $T(\mathbf{c}) \in \mathbb{C}^6$ is defined in [5, 6] as

$$T(\mathbf{c})[k] = \mathbf{w}_*[k] \sum_{n=0}^{N-1} \bar{c}[n] \exp \frac{-j2\pi kn}{N}, \quad 1 \leq k \leq 6 \quad (1)$$

where $\mathbf{w}_* \in \mathbb{R}^6$ is a vector of perceptual weights, and

$$\bar{c}[n] = \frac{c[n]}{\sum_{n=0}^{N-1} c[n]} \quad (2)$$

is used to ensure an amplitude-invariant tonal representation. Equation (1) constitutes a Discrete Fourier Transform (DFT) analyzing the presence of different “frequencies” (i.e., tonal intervals) in the real-valued $\bar{c}$, discarding its DC component (equal to 1 due to the normalization in equation (2)), and weighting the result by $\mathbf{w}_*$.

The motivation for TIVs is the assertion that for two arbitrary chroma vectors $\mathbf{x}$ and $\mathbf{y}$, a distance metric defined in the TIV space (i.e., $\|T(\mathbf{x}) - T(\mathbf{y})\|_2$) corresponds better to the perception of consonance than the same metric defined in the chroma space (i.e., $\|\mathbf{x} - \mathbf{y}\|_2$), due to the combination of normalization and weighting that it applies. The weights were derived from empirical ratings of dyad consonance used to adjust the contribution of each dimension in the space. An initial study determined a weight set for symbolic input as $\mathbf{w}_{\text{symbolic}} = [2, 11, 17, 16, 19, 7]$, using a brute-force search over a set of bounded integers [5]. A follow-up [6] suggested a set of weights for audio input as $\mathbf{w}_{\text{audio}} = [3, 8, 11.5, 15, 14.5, 7.5]$. TIV extraction can be extended to a time-varying chromagram $\mathbf{C} \in \mathbb{R}^{12 \times M}$, applying equation (1) to the chroma vector at each time frame m , yielding $T(\mathbf{C}) \in \mathbb{C}^{6 \times M}$. Alternatively, we can time-average $\mathbf{C}$, yielding the PCP $\mathbf{c}_{\text{PCP}}$ [1] with TIV $T(\mathbf{c}_{\text{PCP}})$.

3. TIV EXTENSIONS

We propose extensions to TIVs in order to support various applications in this work. We express the DFT in equation (1) as a matrix $\mathbf{F}_{\text{complex}} \in \mathbb{C}^{(N/2+1) \times N}$ for even N and indices $0 \leq k \leq N/2$. This matrix has a real analog $\mathbf{F}_{\text{real}} = [\text{Re}(\mathbf{F}_{\text{complex}}); \text{Im}(\mathbf{F}_{\text{complex}})] \in \mathbb{R}^{(N+2) \times N}$. Thus, $\mathbf{F}_{\text{complex}} \mathbf{c} \in \mathbb{C}^{N/2+1}$ and $\mathbf{F}_{\text{real}} \mathbf{c} \in \mathbb{R}^{N+2}$ contain precisely the same information, with the latter stacking the real and imaginary components of the former. We refer to stacking/unstacking of this form as $\text{im2re}/\text{re2im}$, respectively, with $\mathbf{F}_{\text{real}} \mathbf{c} = \text{im2re}(\mathbf{F}_{\text{complex}} \mathbf{c})$ and $\mathbf{F}_{\text{complex}} \mathbf{c} = \text{re2im}(\mathbf{F}_{\text{real}} \mathbf{c})$.

3.1. Inversion

Despite having $N + 2$ rows, $\mathbf{F}_{\text{real}}$ has rank N , as the DC and Nyquist frequency components correspond to purely real basis vectors whose imaginary parts are zero. Discarding the DC component in equation (1) breaks invertibility by removing a rank-contributing row of $\mathbf{F}_{\text{real}}$, reducing the rank of the resulting matrix from N to $N - 1$. We reformulate the transform in a manner that remains equivalent to equation (1) while preserving invertibility. We consider a basis $\mathbf{F} \in \mathbb{R}^{N \times N}$ that starts with $\mathbf{F}_{\text{real}}$, removes its two null components which do not contribute to its matrix rank to create a square matrix, and moves the DC component to the final row of the matrix. By construction, the resulting $\mathbf{F} \in \mathbb{R}^{N \times N}$ has full rank and is therefore invertible. Figure 1 contrasts the different underlying Fourier matrices considered here. Next, we define the augmented vector $\mathbf{w}_{*,2 \times} = [\mathbf{w}_*; \mathbf{w}_{*,:-1}; \kappa]$, where $\mathbf{w}_{*,:-1} = [w_0, \dots, w_{N/2-2}]$ and κ is a free parameter. We construct $\mathbf{W} = \text{diag}(\mathbf{w}_{*,2 \times})$ and the transform $\mathbf{T} = \mathbf{W}\mathbf{F}$. With $\kappa = 0$, equation (1) is recovered exactly via $T(\mathbf{c}) = \text{re2im}(\mathbf{T}\bar{\mathbf{c}})$. This equivalence arises from a reparameterization in which the zero-valued component in equation (1)—previously enforced by the structure of the Fourier basis—is now enforced via $\kappa = 0$.

For $\kappa = 0$, the TIV transform itself is not invertible, but its basis $\mathbf{F}$ is. This fact alone motivates a new PTFR that retains the perceptual smoothing effect of TIV while remaining aligned with the musical pitch axis (see Section 3.4). With $\kappa = 1$, we effectively retain the DC component of the input, whose value is equal to 1 by construction for $\bar{c}$ but is not necessarily so if we also allow ourselves to apply the transform to $\mathbf{c}$ directly. In this setting, the generalized TIV transform becomes invertible, enabling inverse filtering via spectral division to recover the underlying chroma vector. We observe that doing so reveals inconsistencies in the `TIV.lib` implementation [6] of the Temperley profiles [11], as illustrated in Figure 2. While the Shaath profiles encode major and minor scale templates and their transpositions, the recovered Temperley profile exhibits an unexpected emphasis on D in the case of C major. This behavior may explain the degraded performance of these profiles as templates for key detection in our internal evaluations.

3.2. Normalization

We propose optional normalization steps applied to $\mathbf{F}$ and $\mathbf{w}_*$ (i.e., $\mathbf{W}$). Although $\mathbf{F}$ defines an invertible, purely real Fourier basis with orthogonal rows, these rows are not orthonormal as constructed, except for vectors with no imaginary components in their corresponding complex representation. We can normalize the rows of $\mathbf{F}$ to make it orthogonal, ensuring $\|\mathbf{F}\mathbf{x} - \mathbf{F}\mathbf{y}\|_2 = \|\mathbf{x} - \mathbf{y}\|_2$,

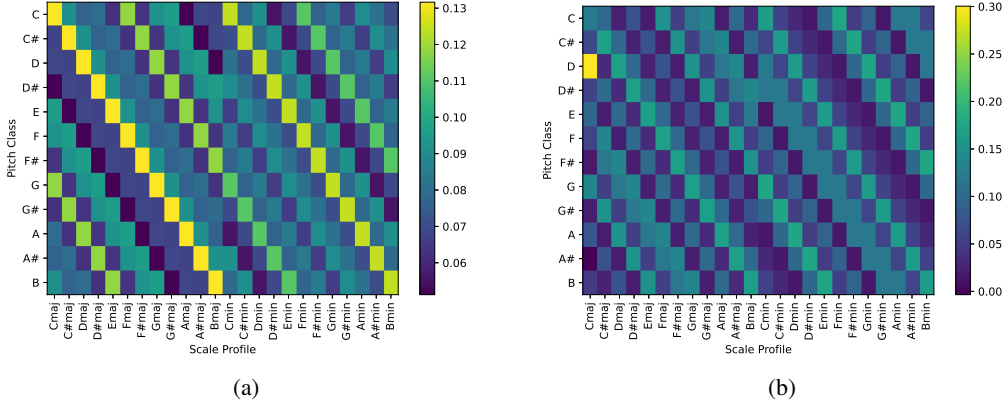


Figure 2: (a) Shaath and (b) Temperley pitch class profiles recovered through our proposed inverse filtering of TIVs provided in `TIV.lib` [6], indicating a potentially problematic C major Temperley profile.

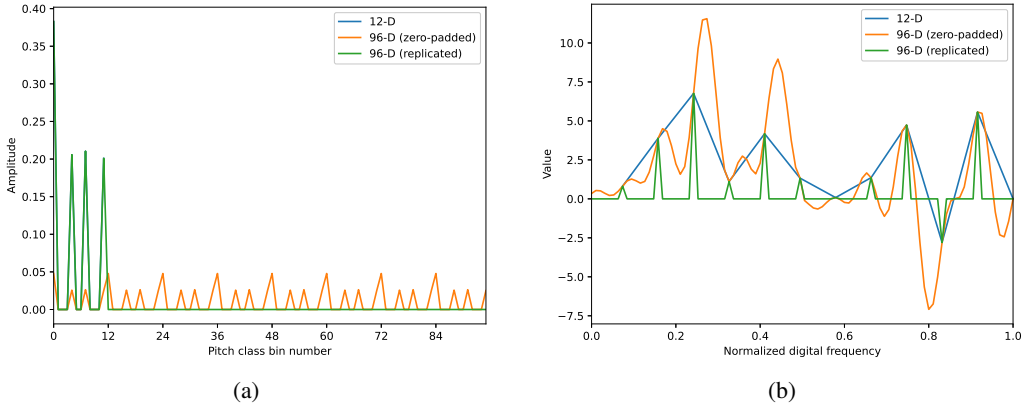


Figure 3: (a) 12- and 96-D PTFRs and (b) their corresponding TIVs using our proposed extensions.

which will be useful in Section 4. We correspondingly rescale the elements of $\mathbf{W}$ to maintain the previously established relationship between TIV distances and empirical consonance ratings in [5].

Moreover, the coefficients in $\mathbf{w}_*$ span the range of $[2, 19]$, which can inflate the magnitude of the TIVs by an order of magnitude (and therefore distance metrics defined in terms of them) relative to their respective chroma vectors. We propose an optional normalization to the weights $\mathbf{w}_{*,1} = [w_1, \dots, w_{N/2-1}]$ such that they sum to $N/2 - 1$, i.e., the sum of the corresponding weights had they been set to unity. This maintains κ as a free parameter while making $\|\mathbf{T}\mathbf{x} - \mathbf{T}\mathbf{y}\|_2 \approx \|\mathbf{x} - \mathbf{y}\|_2$ in the relative sense. Together with the previous normalization steps, this facilitates more interpretable visualization and comparison of the perceptually smoothed PTFR introduced in Section 3.4.

3.3. Higher-dimensional PTFRs

Next, we extend the TIV space defined strictly for $N = 12$ to any $N = 12o$, where o is a non-negative integer indicating the number of octaves captured by a given PTFR. Such an extension analyzes tonal interval patterns extending beyond a single octave, foregoing octave equivalence captured by a chroma vector over when $o > 1$.

Our construction of invertible, optionally orthogonal $\mathbf{F}$, and our optional normalizations scale naturally to 12o-D PTFRs since they are generally expressed in terms of N . Next, we must extend a baseline $\mathbf{w}_*$ for arbitrary $o > 1$. Since a 12o-point transform

increases its tonal interval resolution by a factor o relative to its 12-point counterpart, we opt to oversample a canonical $\mathbf{w}_*$ using linear interpolation. The boundedness and monotonicity properties of linear interpolation ensure that the resulting coefficients remain non-negative. The tonal interval basis frequencies for $N = 12$ will also be present precisely for any $N = 12o$, with their corresponding weights preserved under our implementation. When we consider non-zero values for κ , a design choice arises. One natural approach is to interpolate the weights between κ and their canonical $\mathbf{w}_{*,1}$ across the oversampling interval. Alternatively, we can interpolate them between 0 and their canonical $\mathbf{w}_{*,1}$, followed by the explicit assignment $\mathbf{w}_{*,0} \leftarrow \kappa$. We adopt the latter to remain consistent with the original DC-invariant TIV formulation, while enabling optional DC preservation (see Section 3.4).

Figure 3 demonstrates our proposed extension for computing TIVs over higher-dimensional PTFRs (i.e., with $N = 96$, $o = 8$), omitting our normalization steps for visual clarity. Figure 3a illustrates a 12-D chroma vector, alongside 96-D versions realized in two distinct ways: 1) *zero-padding* the chroma vector with $12(o - 1)$ zeros, and 2) scaling the chroma vector by $1/o$, and *replicating* the result an additional $(o - 1)$ times (i.e., performing a convolution with a dilated averaging kernel). Figure 3b plots the corresponding (real/imaginary-stacked) TIV representations. Both 96-D constructions yield identical values at the frequency indices corresponding to the original chroma TIV, demonstrating that the extension preserves the perceptual weighting defined over chroma

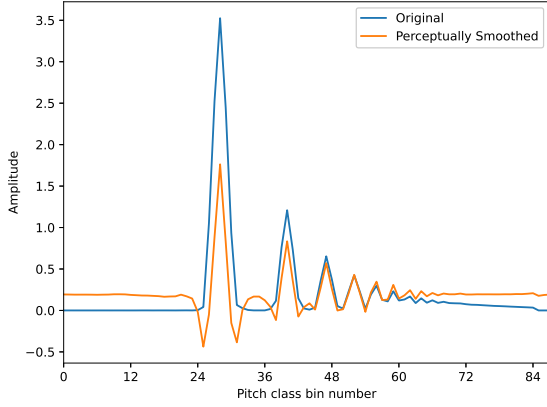


Figure 4: 88-D PTFR and its perceptually smoothed version.

vectors. The replicated version yields zero at all other frequencies due to the Fourier transform properties of periodic signals. This can be interpreted as octave folding acting as a blurring operation in the higher-dimensional PTFR directly. Lastly, the zero-padded version naturally convolves said spectrum with a sinc function.

3.4. Perceptually smoothed PTFRs

Lastly, we define the feature extractor $A(\cdot)$ as

$$A(c) = c' = F^T T c = F^T W F c \quad (3)$$

where we recover $c' = c$ for $w_* = 1$ and $\kappa = 1$. Equation (3) defines a viable 12o-D PTFR, embedding perceptual consonance priors while admittedly no longer guaranteeing the non-negativity of c' , as is the case for c . This representation will prove crucial in the development of our PTFR alignment algorithm in Section 4. We can adapt our operator to any non-12o-D PTFR (e.g., the $N = 88$ case representing the keys of a standard piano keyboard) by appropriately zero-padding it to the nearest 12o, applying our perceptually smoothed PTFR processing, and truncating the result to its initial dimensionality. It also scales to time-varying PTFR matrices of dimension $N \times M$. Figure 4 summarizes the culmination of all of our TIV extensions, illustrating an 88-D transcription-like vector alongside its perceptually smoothed version computed using the inversion, normalization, and feature extraction techniques proposed in this section. The resulting processed version looks like its original PTFR, albeit filtered by weights in the TIV space. Since formulating our method, we observe a structural similarity between our operator and data-driven Fourier Neural Operators (FNO) [12]. The distinction is that our approach defines a fixed, perceptually grounded operator with diagonal, non-negative weights from perceptual consonance data, whereas FNOs learn broader transformations from data with respect to some objective.

4. ALIGNMENT OF PTFRS

Given the extensions proposed in Section 3, we are interested in deriving a matrix $P \in \mathcal{P}_N$ capable of aligning one non-negative PTFR x to another non-negative PTFR y , where $\mathcal{P}_N$ is the set of $N \times N$ permutation matrices. The permutation constraint maintains $\|Pc\|_p = \|c\|_p \forall c, 0 \leq p \leq \infty$, where we are interested in the $p = 0$ case to ensure that pitch classes are not created or

removed by its application. We frame our task as the optimization

$$\min_{P \in \mathcal{P}_N} J(P) \quad (4)$$

$$J(P) = \|TPx - Ty\|_2^2 \quad (5)$$

Note that P is applied to x along the musical pitch axis (i.e., the “time” domain), while the objective function is defined in the TIV space (i.e., the “spectral” domain). Optimizations of this form are common in audio signal processing [13], with iterative solutions often employing gradient updates derived via Wirtinger calculus. The permutation matrix constraint in our case naturally complicates carrying out such an optimization, while the use of a real spectral basis F over real-valued inputs avoids the need for complex analysis techniques used in the aforementioned works.

Expanding the norm of $J(P)$, we arrive at

$$J(P) = \underbrace{(Px)^T T^T T (Px)}_{\text{Quadratic Term}} - 2 \underbrace{(Px)^T T^T Ty}_{\text{Cross Term}} + \underbrace{y^T T^T Ty}_{\text{Constant Term}} \quad (6)$$

With $W = I$, the quadratic term reverts to a constant $\|x\|_2^2$. The resulting optimization would amount to solving a Linear Assignment Problem (LAP) [14], whose solution we will expand on in Section 4.2. However, since W is a non-uniform diagonal weight matrix, $T^T T$ is a full matrix that will not necessarily commute with P , resulting in an NP-hard Quadratic Assignment Problem (QAP) [15]. This amounts to saying that the act of perceptual filtering is dependent on the permutation which we apply onto a PTFR, since it directly dictates the tonal intervals that are present.

Since F is orthogonal, with $\kappa = 0$ we can express $J(P)$ as

$$J(P) = \|TPx - Ty\|_2^2 = \|A(Px) - A(y)\|_2^2 \quad (7)$$

This perspective allows us to consider an intuitive spectral matching solution [16], where we approximate $J(P)$ with

$$J_{\text{LAP}}(P) = \|Px' - y'\|_2^2 = \underbrace{-2(Px')^T y'}_{\text{Cross Term}} + \underbrace{\|x'\|_2^2 + \|y'\|_2^2}_{\text{Constant Terms}} \quad (8)$$

whose minimization involves solving an LAP defined over x' and y' . We incrementally explore different methods for partially or fully carrying out this minimization.

4.1. Circular shifts

We temporarily restrict our attention to the subclass of permutation matrices corresponding to circular shifts, since these transformations admit a direct realization in the audio domain via standard pitch-shifting techniques. In our argmax approach, we consider a circular shift of x' such that $\text{argmax}(Px') = \text{argmax}(y')$. This amounts to defining the musical root of c' as the index of its maximum component (which may be violated in practice in both audio and symbolic domains, but serves as a reasonable prior in the absence of additional information). Since it does not consider the full structure of c' (perceptual filtering aside), this approach often yields results similar to those obtained by applying the method directly to chroma vectors. Basic alignments of this form will be leveraged further in Section 5. The optimal alignment restricted to circular shifts can be obtained via the cross-correlation method [17], a standard technique for estimating relative shifts between signals. This identifies the circular shift that maximizes the similarity between x' and y' , rather than relying solely on their dominant pitch classes. Notably, we apply this method not directly to chroma vectors, but rather to their perceptually smoothed versions.

4.2. Permutations

Broadening our scope to the entire family of permutation matrices, equation (7) admits a global minimizer as the permutation achieving $\text{argsort}(\mathbf{P}\mathbf{x}') = \text{argsort}(\mathbf{y}')$. The argsort method follows from the *rearrangement inequality* [18] stating that the sum of pairwise products between two vectors is maximized when their components are sorted in the same order. As this term appears (with negative sign) as the cross term in the LAP objective, the optimal permutation aligns the order statistics of the two vectors.

Lastly, we extend our alignment approach to time-varying, non-negative PTFRs $\mathbf{X}$ and $\mathbf{Y} \in \mathbb{R}^{N \times M}$, leveraging their perceptually smoothed versions $\mathbf{X}'$ and $\mathbf{Y}'$, respectively. Accordingly, the true QAP and approximate LAP objectives are given by

$$J(\mathbf{P}) = \|\mathbf{TPX} - \mathbf{TY}\|_F^2 = \|\mathbf{A}(\mathbf{PX}) - \mathbf{A}(\mathbf{Y})\|_F^2 \quad (9)$$

$$J_{\text{LAP}}(\mathbf{P}) = \|\mathbf{PX}' - \mathbf{Y}'\|_F^2 = \underbrace{-2\langle \mathbf{PX}', \mathbf{Y}' \rangle}_{\text{Cross Term}} + \underbrace{\|\mathbf{X}'\|_F^2 + \|\mathbf{Y}'\|_F^2}_{\text{Constant Terms}} \quad (10)$$

respectively. The latter is identified as a LAP with cost matrix

$$C_{ij} = -2\langle \mathbf{X}'_j, \mathbf{Y}'_i \rangle \quad (11)$$

which can be solved optimally using the Hungarian algorithm [14]. The Hungarian method could also be used to minimize equation (7) defined over vectors, but it will converge to the same solution given by the rearrangement inequality. This can be understood through the Monge structure [19] of the induced cost matrix, guaranteeing the order-preserving quality in this case. In contrast to the vector case, the matrix case captures time-varying interactions between $\mathbf{X}'$ and $\mathbf{Y}'$ as the assignment is determined by similarities between entire feature vectors rather than a single scalar summary.

4.3. Iterative refinement to solve the QAP

Until now, we have avoided solving the QAP, diverting our attention towards solving the related but inexact LAP. Meanwhile, we find that solving the LAP accounts for a good majority of the potential error improvement in our optimization. We suggest a 2-opt algorithm [20] performing a greedy swap search to refine our initial estimate. Our iterative refinement approach works as follows:

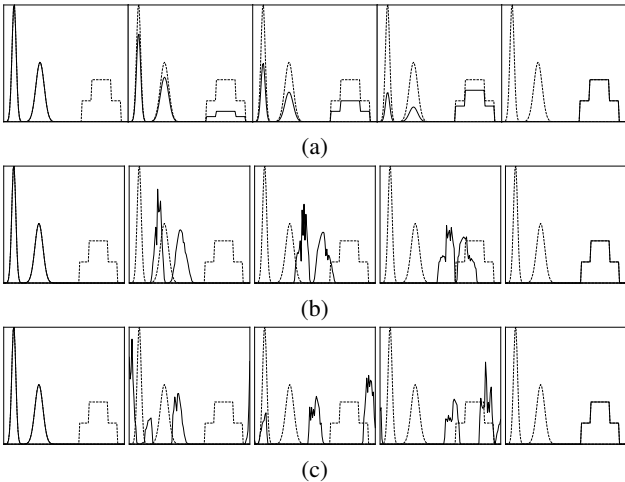


Figure 5: (a) Naive interpolation, (b) linear and (c) circular OT.

Algorithm 1 Computing The Linear OT Plan π^*

```

 $\pi_{i,j}^* \leftarrow 0$ 
 $\rho_x, \rho_y \leftarrow \bar{\mathbf{x}}[0], \bar{\mathbf{y}}[0]$   $\triangleright \rho$  is the mass left in a bin

loop
  if  $\rho_x < \rho_y$  then
     $\pi_{i,j}^* \leftarrow \rho_x$   $\triangleright$  Assign as much mass as possible

     $i \leftarrow i + 1$   $\triangleright$  Refill the emptied bin
    if  $i \geq N$  then break
     $\rho_y \leftarrow \rho_y - \rho_x$   $\triangleright$  Decrease the capacity of the other

     $\rho_x \leftarrow \bar{\mathbf{x}}[i]$ 

  else
    Symmetric to the case above

```

return π^*

1. Find the permutation matrix $\mathbf{P}$ minimizing $J_{\text{LAP}}(\cdot)$.
2. For every pair of indices (i, j) :
 - (a) Tentatively swap the rows $\mathbf{P}_i$ and $\mathbf{P}_j$ to yield $\mathbf{P}^{\text{new}}$.
 - (b) Perform the update $\mathbf{P} \leftarrow \mathbf{P}^{\text{new}}$ if the loss improves.
3. Repeat until no single swap reduces $J(\cdot)$.

Since we are primarily interested in optimizing $\mathbf{P}$ for small N (i.e., the $o = 1$ case with $N = 12$), this proves to be a pragmatic and efficient way to improve upon our initial estimate.

5. INTERPOLATION OF PTFRS

We aim to musically interpolate between two L_1 -normalized PTFRs $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$. In contrast to the alignment setting, we swap the permutation (i.e., L_0 -preserving) constraint with a relaxed L_1 -preserving one, maintaining the interpolated result as a valid probability vectors. A naive solution takes convex combinations of $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$, which simply fades one set of pitches out while another fades in [9]. To overcome this, we build on the approach of [9], which used OT [21] to achieve an audio effect by interpolating re-assigned spectrogram frames over a crossfade interval. While that work considered a linear frequency axis in an audio processing context, we extend it to arbitrary PTFRs aligned to a log-frequency pitch axis, and further propose a specialized circular formulation for chroma vectors. Figure 5 contrasts these methods, adapting the notional illustration from [9]. Naive interpolation corresponds to a linear fade effect, while the displacement method in [9] yields a more intuitive “earth-moving” effect. Our circular formulation utilizes both directions around the musical pitch class circle.

5.1. Linear OT

The OT problem seeks the most efficient way to transform one probability distribution into another. In this setting, distributions are interpreted as mass distributions, and an OT plan π^* describes the minimum work required to move that mass from a source $\bar{\mathbf{x}}$ to a target $\bar{\mathbf{y}}$. Viewing the pitch axis as discrete points in 1D, the OT plan is obtained via monotone rearrangement using the greedy algorithm in Algorithm 1 (directly adapted from [9]). We refer to

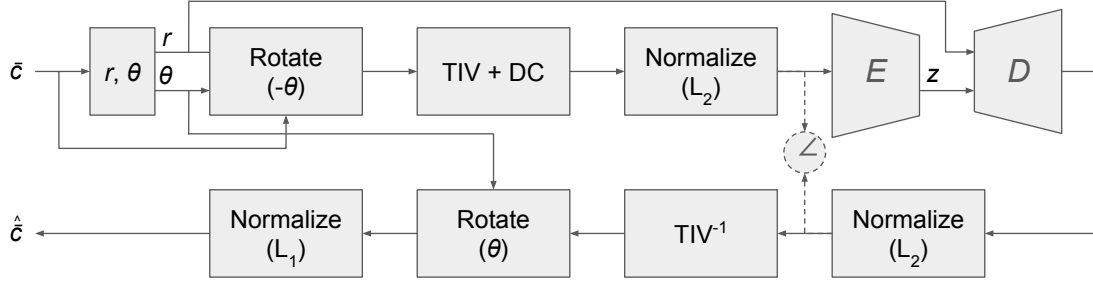


Figure 6: Block diagram of our geometric scale representation model architecture.

Algorithm 2 Circular OT Interpolation For (π^*, t)

Given π^*, t
 $z \leftarrow \mathbf{0}$ $\triangleright z$ is the interpolated result

for $(i, j) \in \text{supp}(\pi^*)$ **do**
 $\Delta_{\rightarrow}, \Delta_{\leftarrow} \leftarrow (j - i) \bmod N, (i - j) \bmod N$
if $\Delta_{\rightarrow} < \Delta_{\leftarrow}$ **then**
 $\omega \leftarrow (i + t\Delta_{\rightarrow}) \bmod N$ $\triangleright$ move clockwise
else
 $\omega \leftarrow (i - t\Delta_{\leftarrow}) \bmod N$ $\triangleright$ move counter-clockwise
 $\omega_1, \omega_2 \leftarrow$ circular integer neighbors of ω
 $z[\omega_1], z[\omega_2] \leftarrow$ fractional split of π_{ij}^*

return z

this as “linear OT” in contrast to the circular OT variant we introduce in Section 5.3. The PTFR can be interpolated with parameter $t \in [0, 1]$ by placing each mass π_{ij}^* at

$$(1 - t)\omega_i^x + t\omega_j^y \quad (12)$$

5.2. Circle of Fourths/Fifths (CoF)

The CoF [10] provides a representation in which harmonic distances between pitch classes are better captured by the number of fourth/fifth relationships separating them, rather than their chromatic distance. Mapping a chroma vector from the chromatic space to the CoF space is defined by yet another permutation matrix $\mathbf{H} \in \mathbb{R}^{12 \times 12}$ with inverse $\mathbf{H}^T$. Applying $\mathbf{H}$ is summarized by $\text{c_cof} = \text{c}[0, 7, 2, 9, 4, 11, 6, 1, 8, 3, 10, 5]$. The circular OT method in Section 5.3 is invariant to whether clockwise/counterclockwise directions correspond to fourths/fifths.

5.3. Circular OT along the CoF

When considering OT-like interpolation over chroma vectors, we incorporate two modifications. Rather than defining transport along the chromatic axis, the OT plan is constructed within the CoF domain. Moreover, we use a circular distance cost, connecting the endpoints (i.e., C and B) to enable transport over a periodic domain. As a result, the circular OT plan (readily computed using the Python OT library [22]) can move mass in either direction around the pitch-class circle. Given the circular OT plan, interpolation is carried out using the greedy algorithm in Algorithm 2 assessing

the shortest-path transfer of mass for each non-zero element π_{ij}^* . When interpolating between two chroma vectors, it is instructive to distinguish between interpolation respecting musical roots and one that captures their underlying scale color independent of root. In the latter case, we may first align one or both chroma vectors to a common root using the argmax method described in Section 7.1 before applying the circular OT interpolation.

6. GEOMETRIC SCALE REPRESENTATION

The alignment procedure in Section 4 enforces a discrete transformation of one chroma vector onto another under a strict L_0 -norm preservation constraint. The interpolation procedure in Section 5 induces a continuous transformation under a relaxed L_1 -norm preservation constraint. Accordingly, we introduce a scale representation learned over chroma vectors enabling interpolation along distinct dimensions of musical structure (scale root, color, and density). We are particularly interested in interpolation along the color axis while preserving the underlying scale density. Beyond providing a chroma compression effect, the approach bridges the gap between alignment and interpolation, enabling continuous musical interpolation under structured density considerations.

We formulate an autoencoder enforcing a cylindrical, 3-D encoding, as illustrated in Figure 6. Of its 3 dimensions, the r and θ dimensions are calculated deterministically, while the z dimension is learned given access to r . The latter is computed as the Hoyer sparsity metric [23], scaled such that a chroma vector with a single note active maps to $1/12$, while a uniform chromatic scale maps to 1. The variable θ represents the angular distance between the determined root from C along the CoF, using the strict mathematical definition of the root as $\text{argmax}(c)$. Inspired by [24], we leverage *transposition-invariant training* [25], whereby we pre-rotate chroma vectors according to $-\theta$ (such that our model only observes chroma vectors with C as their root), followed by a post-rotation by θ following the core neural network processing.

For a given pre-rotation of $\bar{c}$, the input to the model encoder E is the L_2 -normalized, stacked TIV representation with DC preservation (i.e., $\kappa = 1$), denoted $(T\bar{c})_{L_2}$. The encoder outputs a latent variable z , which is concatenated with r and passed to the decoder D , producing a reconstruction $(\widehat{T\bar{c}})_{L_2}$. Model optimization minimizes the cosine distance between the input $(T\bar{c})_{L_2}$ and $(\widehat{T\bar{c}})_{L_2}$. To recover its chroma, the reconstructed TIV is inverted and the resulting profile is rotated back to its original root by means of θ . A final L_1 normalization yields $\hat{c}$. Both E and D are composed of 3 linear layers with a hidden dimensionality of 2, interleaved with LeakyReLU activations.

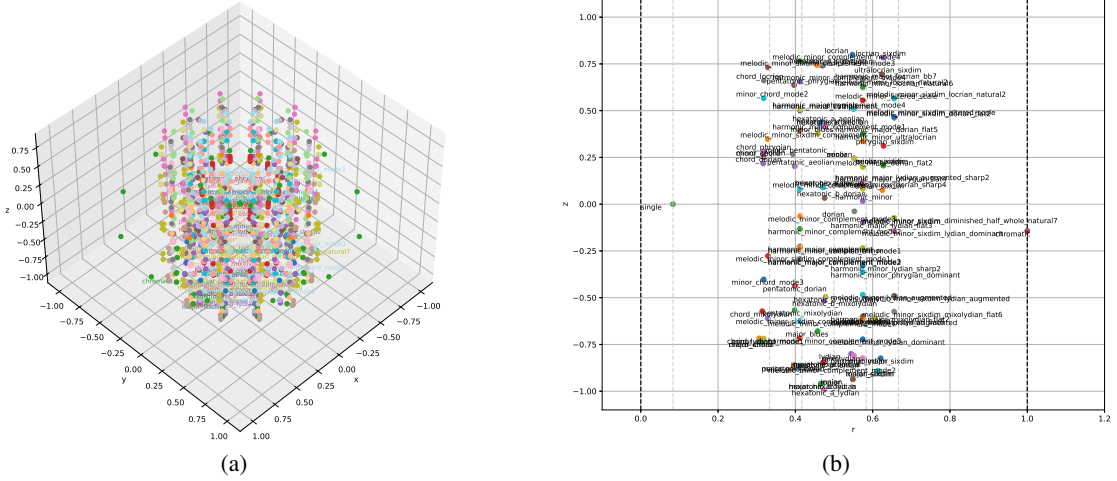


Figure 7: (a) Cylindrical scale representation and (b) rectangular planar profile for $\theta = 0$.

Table 1: Sequence alignment results, with random columns reporting averages over 1024 randomly generated chromagram pairs.

Method	$\ T(\mathbf{P}\mathbf{x}_{\text{PCP}}) - T(\mathbf{y}_{\text{PCP}})\ _2$		$\ T(\mathbf{P}\mathbf{X}) - T(\mathbf{Y})\ _F$	
	Musical	Random	Musical	Random
None	23.354	7.308	130.450	26.344
argmax	13.206	6.534	104.226	24.174
Cross-correlation	13.206	5.055	104.226	20.277
argsort	5.361	3.086	100.266	15.964
argsort+2-opt	4.350	2.276	107.568	14.509
Hungarian	7.622	3.183	90.537	14.889
Hungarian+2-opt	7.891	2.559	89.847	13.012

7. EXPERIMENTAL RESULTS

Due to space constraints, we refer readers to additional figures and audio demos available at <https://tiv-ext.netlify.app>.

7.1. Alignment of PTFRs

To illustrate our proposed alignment approach, we consider a simple example consisting of two synthetically generated *musical* signals. The chromagram $\mathbf{X}$ represents a I-IV-V-I chord sequence in D major, with slight perturbations applied to its weights for realism. The chromagram $\mathbf{Y}$ represents the standard minor cantus firmus 1-5-4-b3-2-1-b3-2-1 [26] in C minor. We derive permutations using the methods outlined in Section 4. For methods operating over a single chroma vector, we compute PCPs $\mathbf{x}_{\text{PCP}}$ and $\mathbf{y}_{\text{PCP}}$ for $\mathbf{X}$ and $\mathbf{Y}$, respectively. The resulting permutation is then applied to $\mathbf{X}$. In order to demonstrate our alignment approach at scale, we also create a *random* dataset of 1024 randomly generated pairs of $\mathbf{X}$ and $\mathbf{Y}$ chromagrams to assess average optimizer performance.

Alignment results are summarized in Table 1, illustrating the incremental improvements incurred between methods. In order to avoid biasing results towards our own creations, distance metrics are reported given the original TIV in equation (1). We see that the argsort and Hungarian methods yield the best results for the LAP approximation for the vector and matrix cases, respectively. The proposed 2-opt refinement can improve upon these solutions by solving the true QAP. Though less expressive, shift-only methods

Table 2: Interpolation between C major and C minor triads.

t	Linear Fade / OT		Circular OT along CoF	
	Note Set	Chord	Note Set	Chord
0	[C, E, G]	C	[C, E, G]	C
1/3	[C, D $\sharp$, E, G]	Cadd $\sharp 9$	[C, F, G, A]	Fadd9
1/2	[C, D $\sharp$, E, G]	Cadd $\sharp 9$	[C, D, F, G, A, B $\flat$]	Gm9 / C
2/3	[C, D $\sharp$, E, G]	Cadd $\sharp 9$	[C, D, G, B $\flat$]	Gm / C
1	[C, E $\flat$, G]	Cm	[C, E $\flat$, G]	Cm

still incrementally improve upon initial alignments.

7.2. Interpolation of PTFRs

To demonstrate our proposed interpolation approach, we consider blending between $\bar{\mathbf{x}} = [1/3, 0, 0, 0, 1/3, 0, 0, 1/3, 0, 0, 0]$ (C major) and $\bar{\mathbf{y}} = [1/3, 0, 0, 1/3, 0, 0, 0, 1/3, 0, 0, 0]$ (C minor). We evaluate the resulting interpolation for $t = [0, 1/3, 1/2, 2/3, 1]$ achieved by naive linear interpolation (fading), linear OT displacement, and our proposed circular OT in the CoF domain. The baseline methods converge to the same solution in this case since the dominant transport occurs from E to D $\sharp$ /E $\flat$. While these methods produce musically unstructured fades, the proposed interpolation outlines a plausible sequence of harmonic modulations, as summarized in Table 2. At $t = 0.5$ (the Wasserstein barycenter [27]), the interpolation yields a “colorless” hexatonic note set that is neither major nor minor, corresponding to the largest subset of the major scale or its modes exhibiting this property. Its invariance under reflection about the *negative harmony axis* [10] highlights its intrinsic harmonic balance.

7.3. Geometric scale representation

We train our representation on a set of 106 internally constructed scale profiles. The profiles span tetrachord, pentatonic, hexatonic, heptatonic, and octatonic scales, as well as a chromatic and “single-note” scale with one active pitch class. Weights are derived from the Krumhansl–Schmuckler and Temperley profiles [11] setting non-scale tones weights to zero. Each training step uses the full training set as a batch. The model is trained for 20K steps using the Adam optimizer with a learning rate of 10^{-3} .

Figure 7 shows our learned encodings over the scale set. We center z using its value corresponding to the “single-note” scale and rescale its distribution to lie within $[-1, 1]$. We observe Locrian-like and Lydian-like profiles at the extremes of z , indicating that it captures a spectrum of bright and dark tonalities. Scales are organized by density along the orthogonal r axis. We demonstrate smooth aural transitions of decoded scale profiles along the scale-color axis by varying z while holding r fixed on our demo website.

8. CONCLUSIONS

We presented a framework for modeling pitch structure in audio and symbolic music through perceptually grounded transformations of PTFRs. By extending the TIV into an invertible and flexible operator, we enabled a new family of PTFRs and leveraged its properties for aligning musical structures. We introduced a circular OT-based interpolation that yields musically meaningful transitions between pitch distributions, consistent with principles of musical voice leading. Finally, we introduced a learned geometric scale representation that combines these perspectives to create an interpretable structure in terms of scale color and density. These contributions demonstrate that perceptually informed signal processing techniques can yield interpretable models of musically pitched material, offering a foundation for structure-aware audio analysis and transformation. Future work will explore alternative alignment objectives (e.g., cosine or L_1 distances) and investigate the integration of our feature extractor within the circular OT framework, which is currently constrained by OT’s non-negativity requirement. We also aim to further develop the geometric scale representation by introducing a uniform, bounded bottleneck for z (e.g., via a variational autoencoder with a truncated Gaussian prior [28] and/or finite scalar quantization [29]). Finally, we plan to extend these ideas toward enabling deeper and more structured control of PTFR-conditioned generative music systems.

9. REFERENCES

- [1] E. Gómez, “Tonal description of polyphonic audio for music content processing,” *INFORMS Jour. Comp.*, vol. 18, no. 3, pp. 294–304, 2006.
- [2] R. Bittner *et al.*, “A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation,” in *Proc. IEEE Intl. Conf. Ac., Sp. Sig. Proc.*, 2022.
- [3] M. Müller, *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Springer, 2015.
- [4] D. Ellis and G. Poliner, “Identifying ‘cover songs’ with chroma features and dynamic programming beat tracking,” in *Proc. IEEE Intl. Conf. Ac., Sp. Sig. Proc.*, 2007.
- [5] G. Bernardes *et al.*, “A multi-level tonal interval space for modelling pitch relatedness and musical consonance,” *Jour. New Mus. Res.*, vol. 45, no. 4, pp. 281–294, 2016.
- [6] A. Ramires, G. Bernardes, M. Davies, and X. Serra, “TIV.lib: An open-source library for the tonal description of musical audio,” in *Proc. Intl. Conf. Dig. Aud. Eff.*, 2020.
- [7] J. Copet *et al.*, “Simple and controllable music generation,” in *Proc. Conf. Neur. Inf. Proc. Sys.*, 2023.
- [8] J. Nistal *et al.*, “Diff-A-Riff: Musical accompaniment co-creation via latent diffusion models,” in *Proc. Intl. Soc. for Mus. Inf. Ret. Conf.*, 2024.
- [9] T. Henderson and J. Solomon, “Audio transport: A generalized portamento via optimal transport,” in *Proc. Intl. Conf. Dig. Aud. Eff.*, 2019.
- [10] E. Levy, *A Theory of Harmony*. State University of New York Press, 1985.
- [11] D. Temperley, “What’s key for key? the Krumhansl-Schmuckler key-finding algorithm reconsidered,” *Music Perception*, vol. 17, no. 1, pp. 65–100, 1999.
- [12] Z. Li *et al.*, “Fourier neural operator for parametric partial differential equations,” in *Proc. Intl. Conf. on Learn. Rep.*, 2021.
- [13] H. Caracalla and A. Roebel, “Gradient conversion between time and frequency domains using Wirtinger calculus,” in *Proc. Intl. Conf. Dig. Aud. Eff.*, 2017.
- [14] H. Kuhn, “The Hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [15] T. Koopmans and M. Beckmann, “Assignment problems and the location of economic activities,” *Econometrica*, vol. 25, no. 1, pp. 53–76, 1957.
- [16] B. M. Levitan, *Inverse Sturm-Liouville Problems*. VNU Science Press, 1987.
- [17] A. Oppenheim, A. Willsky, and S. Nawab, *Signals and Systems*. Prentice Hall, 1999.
- [18] G. Hardy, J. Littlewood, and G. Pólya, *Inequalities*, 2nd ed. Cambridge University Press, 1952.
- [19] R. Burkard, M. Dell’Amico, and S. Martello, *Assignment Problems*. SIAM, 2009.
- [20] G. Croes, “A method for solving traveling-salesman problems,” *Operations Research*, vol. 6, no. 6, pp. 791–812, 1958.
- [21] C. Villani, *Optimal Transport: Old and New*. Springer, 2009.
- [22] R. Flamary *et al.*, “POT: Python optimal transport,” *Jour. Mach. Learn. Res.*, vol. 22, no. 78, pp. 1–8, 2021.
- [23] P. Hoyer, “Non-negative matrix factorization with sparseness constraints,” *Jour. Machine Learning Research*, vol. 5, no. Nov, pp. 1457–1469, 2004.
- [24] Y. Wu, “Self-supervised disentanglement of harmonic and rhythmic features in music audio signals,” in *Proc. Intl. Conf. Dig. Aud. Eff.*, 2023.
- [25] A. Arzt and S. Lattner, “Audio-to-score alignment using transposition-invariant features,” in *Proc. Intl. Soc. for Music Inf. Ret. Conf.*, 2018.
- [26] J. Fux, *The Study of Counterpoint: From Johann Joseph Fux’s Gradus ad Parnassum*. W. W. Norton & Company, 1965.
- [27] M. Agueh and G. Carlier, “Barycenters in the Wasserstein space,” *SIAM Jour. Math. Anal.*, vol. 43, no. 2, pp. 904–924, 2011.
- [28] Q. Zhao, N. Honnorat, E. Adeli, and K. M. Pohl, “Truncated gaussian-mixture variational autoencoder,” *arXiv preprint arXiv:1902.03717*, 2019.
- [29] F. Mentzer, D. Minnen, E. Agustsson, and M. Tschannen, “Finite scalar quantization: VQ-VAE made simple,” in *Proc. Intl. Conf. Learn. Rep.*, 2024.