A Model You Can Hear: Audio Classification with Playable Prototypes

Romain Loiseau[1, 2] romain.loiseau@enpc.fr
Baptiste Bouvier[3] baptiste.bouvier@ircam.fr
Yann Teytaut[3] yann.teytaut@ircam.fr
Elliot Vincent[1,4] elliot.vincent@enpc.fr
Mathieu Aubry[1] mathieu.aubry@enpc.fr
Loïc Landrieu[2] loic.landrieu@ign.fr

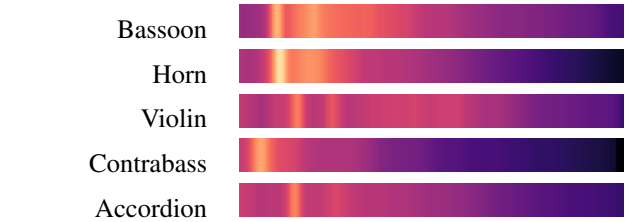
[1]LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, France
[2]LASTIG, Univ Gustave Eiffel, IGN, ENSG
[3]STMS Lab, UMR 9912 (IRCAM, CNRS, Sorbonne University), Paris, France
[4]Inria and DIENS (ENS-PSL, CNRS, Inria)

Abstract

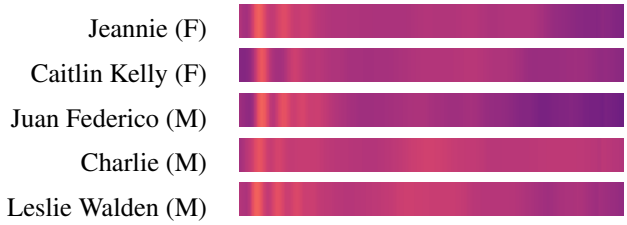
Machine learning techniques have proved useful for classifying and analyzing audio content. However, recent methods typically rely on abstract and high-dimensional representations that are difficult to interpret. In this paper, we propose adapting the transformation-invariant clustering paradigm–which has shown impressive results for both image and 3D data–to the audio domain in a supervised setting. This results in an audio classification model based on prototypical sounds that can be heard directly. Evaluated on speaker and instrument identification tasks, our method produces state-of-the-art results while remaining easily interpretable.

1. Introduction

The emergence of deep learning approaches dedicated to audio analysis has led to significant performance improvements [1, 15]. Although these methods take sound clips as input, they typically rely on a latent space of high dimension, making the interpretation of their decisions difficult and limiting the insights gained on the data and the considered task. Deep learning-enabled transformation-invariant clustering [10, 12, 13] is a recent method in which decisions are based on a small collection of prototypes existing in the same space as the input samples, *e.g.* images or 3D point clouds. Each prototype learns a limited set of transformations in the manner of Spatial Transformation Networks [7], allowing them to approximate a rich but consistent portion of a sample collection. We propose adapting this approach to the audio domain, which results in both high classification accuracy and improved interpretability. Our model is trained in a supervised setting by tasking each prototype to minimize the reconstruction error for the samples of an assigned class. We also propose to explicitly consider in the loss function a soft class assignment based on the reconstruction error.



(a) Instrument dataset SOL [3, 6]



(b) Speech dataset Librispeech [16]

Figure 1. **Playable Prototypes.** Our model consists of a set of spectral prototypes that can automatically adjust their pitch and amplitude to reconstruct input samples. We show such prototypes learned from the SOL [3, 6] and Librispeech [16] datasets.

2. Related Work

Audio Classification. Musical instrument and speaker identification are two of the standard tasks used to benchmark audio classification models. While early musical instrument identification methods focused on distinguishing

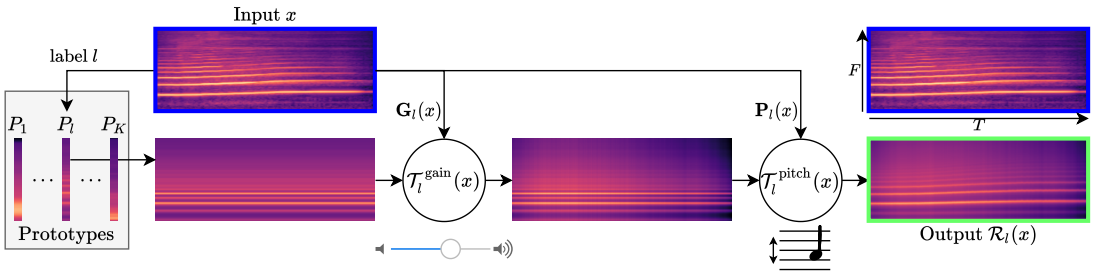


Figure 2. **Method overview.** Given an input sound, we predict for each prototype a *gain* and *pitch* shift at each time stamp to generate the output. Prototypes and transformations are learned jointly using both reconstruction and classification losses.

individual tones achieved appreciable precision, support vector machine operating on spectro-temporal sound representations can reach almost perfect accuracy. As shown in [4], the supervised classification of instruments playing individual notes could be considered solved [11]. Similarly, speaker identification is mostly handled by convolutional and/or recurrent neural networks [2], and also achieve almost perfect performances [21]. However, these models rely on complex and abstract latent representations that are not easily—if at all—interpretable, and cannot be visualized as spectrograms or heard in the audio domain.

Prototype-Based Methods. Auto-Encoders [17,20] learn a compact latent representation and are supervised by a reconstruction task. Through regularization(s) and/or constraint(s), the latent space can be made fit for various tasks such as classification [14]. Yet, despite this controllable structure, the learned features remain mostly non-interpretable. Inspired by the literature on images [9], the authors of [22] propose to generate prototypes from the latent space with a frequency-dependent similarity measure between the prototypes and the latent code. This similarity can then be used for speech, music, and environmental sound classification. However, since the prototypes are not in the audio domain, their interpretability remains limited.

Transformation-Invariant Modeling. Deep transformation invariant clustering [12] learns explicit prototypes in input space. Each prototype is equipped with dedicated transformation networks, allowing a small set of prototypes to faithfully represent a large collection of samples. The resulting models can be used for downstream tasks such as classification [12], few-shot segmentation [10], and even multi-object instance discovery [13]. Jaderberg *et al.* [7] also proposes to learn differentiable transformations in input space, but applies them to the input instead of learned prototypes. We propose to extend these ideas to the audio domain by learning prototypical spectrograms along with adapted transformations.

3. Method

We consider a set of N audio clips $x_1, \cdots, x_N$, each characterized by their log-mel-spectrogram with T time steps and F mel-frequency bins. The samples are annotated with class labels $l_1, \cdots, l_N$. Our goal is to learn a set of K interpretable prototypes in spectral space to explore datasets and perform classification. We equip each prototype with a set of dedicated transformation networks, that are jointly trained to reconstruct audio samples from given classes. The predicted transformations affect specific characteristics of the prototypes, such as amplitude and pitch, allowing for the faithful approximation of a wide variety of samples by each prototype. As a consequence, prototypes can learn to represent more meaningful audio attributes, such as timbre or intonations.

3.1. Deep Transformation-Invariant Prototyping

We learn a set of K prototypes $P_k \in \mathbb{R}^F$, each associated with two transformation networks $\mathcal{T}_k^{\text{gain}}$ and $\mathcal{T}_k^{\text{pitch}}$. Each transformation $\mathcal{T}_k$ takes an audio sample x as input and predicts a transformation in the spectral domain for each time step t of the input. The resulting transformations are then applied sequentially to the prototype P_k for each time step to define the reconstruction $\mathcal{R}_k(x)$:

$$\mathcal{R}_k(x)[t] = \mathcal{T}_k^{\text{pitch}}(x)[t] \circ \mathcal{T}_k^{\text{gain}}(x)[t]\,(P_k)\ , \qquad (1)$$

where $[t]$ denotes the t-th timestamp of a spectrogram, or the spectral transformation to be applied at the t-th time step. We propose to use two different transformations:

- **Amplification.** A given prototype should be able to reconstruct samples independently of their amplitude. We define a transformation $\mathcal{T}_k^{\text{gain}}$ that maps an audio sample x to a gain $\mathbf{G}_k(x)[t] \in \mathbb{R}$ for each t, which we add to all frequencies of the log-mel-spectrogram P_k. This corresponds to adapting the *amplitude* of the prototype to an input sample.
- **Pitch Shifting.** A single prototype is expected to reconstruct samples independently of their pitch. We define $\mathcal{T}_k^{\text{pitch}}$, which produces a pitch shift $\mathbf{P}_k(x)[t] \in [1/2, 2]$ for each time stamp t. We use this value to stretch the spectrogram using linear interpolation along the spectral axis.

Table 1. **Results.** Accuracy and reconstruction error computed on the test sets of SOL [3, 6] and LibriSpeech [16].

	OA	AA	$\mathcal{L}_{\text{rec}}$
SOL [3, 6]			
Direct Classification	97.8	94.8	—
APNet [22]	95.3	91.3	**0.1**
Ours	**99.5**	**96.3**	4.0
LibriSpeech [16]			
Direct Classification	99.4	99.5	—
APNet [22]	97.8	97.8	**0.2**
Ours	**99.9**	**99.9**	3.1

3.2. Loss Functions

Each class is assigned a prototype and its associated transformation networks which are trained to reconstruct all samples from this class. We measure the quality of a reconstruction as the average ℓ_2 distance between the input spectrogram and the transformed prototype for all time steps. For an input audio sample x of class l, we define the following reconstruction loss:

$$\mathcal{L}_{\text{rec}}(x, l) = \frac{1}{T} \sum_{t=1}^{T} \|x[t] - \mathcal{R}_l(x)[t]\|^2 \ . \tag{2}$$

To better encourage the prototypes to discriminate between classes, we propose to jointly optimize the reconstruction loss and the cross-entropy between the true label and the soft minimum of the reconstruction error:

$$\mathcal{L}_{\text{ce}}(x, l) = -\log\left(\frac{\exp\left(-\beta \mathcal{L}_{\text{rec}}(x, l)\right)}{\sum_{k=1}^{K} \exp\left(-\beta \mathcal{L}_{\text{rec}}(x, k)\right)}\right) \ , \tag{3}$$

where β is a learnable parameter corresponding to the inverse temperature in the softmin. To train our network, we use a weighted sum of both losses:

$$\mathcal{L}(x, l) = \sum_{n=1}^{N} \mathcal{L}_{\text{rec}}(x_n, l_n) + \lambda_{\text{ce}} \mathcal{L}_{\text{ce}}(x_n, l_n) \ , \tag{4}$$

with λ_{ce} an hyperparameter set to 0.01. At test time, we can perform classification by predicting for a sample x the prototype that provides the best reconstruction. The models $\mathcal{R}_k$ are trained for classification through the supervision of their reconstruction error. Our method is detailed in Figure 2.

3.3. Parameterization and Training Details

We implement functions $\mathbf{G}$ and $\mathbf{P}$ as one-dimensional convolutional U-Net style networks [18] operating on the temporal dimension. To save parameters, both transformation networks share the same encoder. The prototypes $P_1, \cdots, P_K$ and the inverse temperature β are directly learnable parameters of the model.

We ensure the stability of our model by gradually increasing its complexity along the training procedure. We first learn prototypes without any transformation. After convergence, we sequentially equip each prototype with a gain transformation and pitch-shifting.

We use the ADAM [8] optimizer with a learning rate of 10^{-4}, and a weight decay of 10^{-6} for the transformation networks and 0 for the prototypes and β parameter.

4. Experiments

Datasets. We evaluate our method for classification and reconstruction tasks on the following two datasets:

- **SOL [3, 6].** This dataset contains 24 450 samples of individual notes played with various playing techniques by 33 different instruments and sampled at 44.1kHz. We evaluate instrument classification.
- **Librispeech [16].** This 1000-hour corpus contains English speech sampled at 16kHz. We selected the 128 predominant speakers from the `train-clean-360` set. We evaluate speaker classification.

For both datasets, we randomly selected 70% of the clips for training, 10% for validation, and 20% for testing.

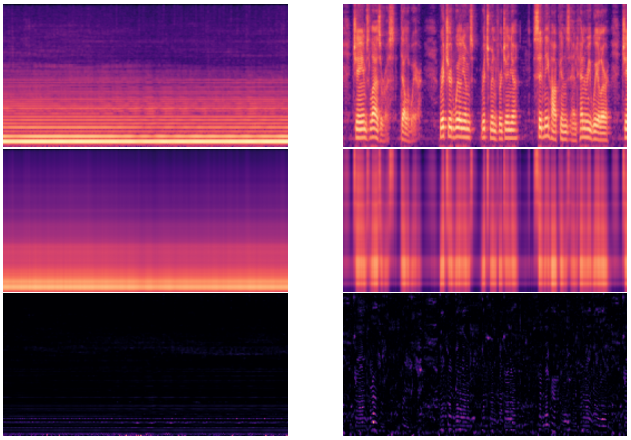
Metrics. To assess the quality of our model, we report the following metrics:

- **Overall Accuracy (OA).** Percentage of input samples correctly classified by our model, *i.e.* best reconstructed by the prototype assigned to their true class.
- **Average Accuracy (AA).** Average of the classwise accuracy, computed across classes without weights.
- **Reconstruction error ($\mathcal{L}_{\text{rec}}$).** To assess the quality of the reconstruction, we also report the reconstruction error $\mathcal{L}_{\text{rec}}$.

Baselines. To put the performance of our method in context, we trained a temporal convolutional network to classify log-mel-spectrograms. We supervise this network with the cross-entropy on the labels. This network, which does not provide a reconstruction error, is called *Direct Classification*. We also evaluated the APNet [22] approach on our datasets. However, this method is limited in its interpretability, as it relies on latent prototypes and uses a fully connected layer to classify samples based on their similarity to the prototypes.

4.1. Quantitative Results

As shown in Table 1, our method reaches near perfect accuracy for both datasets. We observe that APNet [22] reaches better reconstruction scores than our model. This is



(a) SOL [3, 6] (b) Librispeech [16]

Figure 3. **Reconstruction of input clips.** For each input sample (top), we show the reconstruction provided by the model (middle) and the error (bottom). Note how the timbre of the speaker or instrument is well represented, as the model provides reconstructions of the inputs for varying pitches. This leads to an insightful characterization of each instrument or speaker.

expected as APNet uses an encoder-decoder network which can learn rich transformations to and from the latent space. In contrast, we restrict the scope of the spectral transformations to learn prototypes that can only reconstruct meaningfully samples from their assigned class. However, our model is still capable of producing faithful reconstructions, which would be suitable for audio generation tasks (see Sec. 4.2).

4.2. Qualitative Results

Prototypes learn to reconstruct and recognize the origin of audio samples obtained under various conditions: different notes, techniques, words, etc. Furthermore, our model automatically adjusts pitch and amplitude to fit an input sound. Consequently, prototypes learn characteristics of their assigned class that go beyond simple harmonic components (see Figure 3). Instead, our model captures elements of the timbre, such as their spectral envelopes [19], as seen in Figure 1. This opens the way to further tasks such as sound analysis [19] and timbre transfer [5].

5. Conclusion

We presented a new approach to audio understanding by representing large collections of audio clips with few prototypes equipped with learned transformation networks. Our model produces concise, expressive, and interpretable overviews of raw audio clips while retaining state-of-the-art results for audio classification tasks.

6. Acknowledgements

This work was supported in part by ANR project READY3D ANR-19-CE23-0007 and was granted access to the HPC resources of IDRIS under the allocation 2022-AD011012096R1 made by GENCI. We thank Theo Deprelle, Nicolas Gonthier, Tom Monnier and Yannis Siglidis for inspiring discussions and valuable feedback.

References

[1] Jakob Abeßer. A review of deep learning based methods for acoustic scene classification. *Applied Sciences*, 2020. 1

[2] Zhongxin Bai and Xiao-Lei Zhang. Speaker recognition based on deep learning: An overview. *Neural Networks*, 2021. 2

[3] Guillaume Ballet, Riccardo Borghesi, Peter Hoffmann, and Fabien Lévy. Studio online 3.0: An internet "killer application" for remote access to IRCAM sounds and processing tools. In *Journées d'Informatique Musicale*, 1999. 1, 3, 4

[4] DG Bhalke, CB Rao, and Dattatraya S Bormane. Automatic musical instrument classification using fractional fourier transform based-MFCC features and counter propagation neural network. *Journal of Intelligent Information Systems*, 2016. 2

[5] Russell Sammut Bonnici, Charalampos Saitis, and Martin Benning. Timbre transfer with variational auto encoding and cycle-consistent adversarial networks. *arXiv:2109.02096*, 2021. 4

[6] Carmine Emanuele Cella, Daniele Ghisi, Vincent Lostanlen, Fabien Lévy, Joshua Fineberg, and Yan Maresz. OrchideaSOL: a dataset of extended instrumental techniques for computer-aided orchestration. *ICMC*, 2020. 1, 3, 4

[7] Max Jaderberg, Karen Simonyan, and Andrew Zisserman. Spatial transformer networks. *NeurIPS*, 2015. 1, 2

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 3

[9] Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions. In *AAAI*, 2018. 2

[10] Romain Loiseau, Tom Monnier, Mathieu Aubry, and Loïc Landrieu. Representing Shape Collections with Alignment-Aware Linear Models. In *3DV*, 2021. 1, 2

[11] Vincent Lostanlen, Joakim Andén, and Mathieu Lagrange. Extended playing techniques: the next milestone in musical instrument recognition. In *International Conference on Digital Libraries for Musicology*, 2018. 2

[12] Tom Monnier, Thibault Groueix, and Mathieu Aubry. Deep Transformation-Invariant Clustering. In *NeurIPS*, 2020. 1, 2

[13] Tom Monnier, Elliot Vincent, Jean Ponce, and Mathieu Aubry. Unsupervised Layered Image Decomposition into Object Prototypes. In *ICCV*, 2021. 1, 2

[14] Javier Naranjo-Alcazar, Sergi Perez-Castanos, Pedro Zuccarello, Fabio Antonacci, and Maximo Cobos. Open set audio classification using autoencoders trained on few data. *Sensors*, 2020. 2

[15] Ndiatenda Ndou, Ritesh Ajoodha, and Ashwini Jadhav. Music genre classification: A review of deep-learning and traditional machine-learning approaches. In *IEMTRONICS*, 2021. 1

[16] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *ICASSP*, 2015. 1, 3, 4

[17] Fanny Roche, Thomas Hueber, Samuel Limier, and Laurent Girin. Autoencoders for music sound modeling: a comparison of linear, shallow, deep, recurrent and variational models. *Sound and Music Computing*, 2018. 2

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, 2015. 3

[19] Diemo Schwarz. Spectral envelopes in sound analysis and synthesis. Master's thesis, Universität Stuttgart, IRCAM, Paris, 1998. 4

[20] Noé Tits, Fengna Wang, Kevin El Haddad, Vincent Pagel, and Thierry Dutoit. Visualization and interpretation of latent spaces for controlling expressive speech synthesis through audio analysis. *Conference of the International Speech Communication Association*, 2019. 2

[21] Feng Ye and Jun Yang. A deep neural network model for speaker identification. *Applied Sciences*, 2021. 2

[22] Pablo Zinemanas, Martín Rocamora, Marius Miron, Frederic Font, and Xavier Serra. An interpretable deep learning model for automatic sound classification. *Electronics*, 2021. 2, 3