LEARNABLE GROUP TRANSFORM FOR TIME-SERIES

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose to undertake the problem of representation learning for time-series by considering a Group Transform approach. This framework allows us to, first, generalize classical time-frequency transformations such as the Wavelet Transform, and second, to enable the learnability of the representation. While the creation of the Wavelet Transform filter-bank relies on the sampling of the affine group in order to transform the mother filter, our approach allows for non-linear transformations of the mother filter by introducing the group of strictly increasing and continuous functions. The transformations induced by such a group enable us to span a larger class of signal representations. The sampling of this group can be optimized with respect to a specific loss and function and thus cast into a Deep Learning architecture. The experiments on diverse time-series datasets demonstrate the expressivity of this framework which competes with state-of-the-art performances.

1 INTRODUCTION

The selection of the time-frequency representation for analyzing, classifying, and predicting time-series has long been studied (Coifman & Wickerhauser, 1992; Mallat & Zhang, 1993; Gribonval & Bacry, 2003). To this day, the front-end processing of time-series remains a keystone toward the improvement of a wealth of applications such as health-care (Saritha et al., 2008; Cosentino et al., 2016)), environmental sound (Balestriero et al., 2018; Lelandais & Glotin, 2008), and seismic data analysis (Seydoux et al., 2016; Liu & Fomel, 2013). The common denominator of the recorded signals in these fields is their undulatory behavior. While these signals share this common behavior, two significant factors imply the need of learning the representation: **1)** time-series are intrinsically different because of their physical nature, **2)** the machine learning task can be different even within the same type of data. Therefore, the representation should be induced by both the signal and the task at hand.

An all too common approach to performing inference on time-series consists of building a Deep Neural Network (DNN) that operates on a spectral decomposition of the time-series such as Wavelet Transform (WT) or Mel Frequency Spectral Coefficients (MFSC). The selection of the judicious transform is either performed by an expert in the signal at hand, or by considering the aforecited selection methods and their derivatives. However, an inherent drawback is that the selection of the time-frequency transform is often achieved with criteria that do not align with the task. For instance, a selection based on the sparsity of the representation while the task is the classification of the signals. Besides, these selection methods and transformations require substantial cross-validations of a large number of hyperparameters such as mother filter family, number of octaves, and number of wavelets per octave, size of the window (Cosentino et al., 2017).

To alleviate these drawbacks, we present a Learnable Group Transform (LGT) that can be learned jointly with the inference optimization problem and the data at hand. Our methodology is developed through the lens of *harmonic analysis*, which generalizes the Fourier Transform using group theory. We will study the learnability of a *group transform*. Well-known group transforms are the Short-Time Fourier Transform (STFT) and the Continuous Wavelet Transform (CWT). The theoretical building blocks of time-frequency analysis via group transforms are well developed in *Coorbit Theory* Feichtinger & Gröchenig (1989) and *Generalized Coherent States* Grossmann et al. (1985; 1986). In order to build a group transform, one requires to select two elements; a mother filter and a group (Section 2).

The learnability of a mother filter has been already developed in Ravanelli & Bengio (2018); Balestriero et al. (2018); Cakir et al. (2016); Zeghidour et al. (2018). Recently, Khan & Yener (2018) investigated the learnability of the affine transformations, that is, the sampling of dilation parameter of the affine group inducing the CWT. This filter-bank is then used to build a group transform of the signal. In this work, we propose to extend the learnability of the affine transform into strictly increasing and continuous functions enabling non-linear transformations of the mother filter (Section 3).

This generalization allows for greater flexibility in the learnable spectral decomposition. This flexibility improves the linearization capability of the representation as it eases the learning of a spectral decomposition that is able to discard intricate patterns in the time-series that are nuisances. Also, it implies that for fixed network topology, replacing the learnable affine group with the continuous group leads to a larger class of spannable functions which improves the approximation property of the DNN at hand (Winkler & Le, 2017; Balestriero & Baraniuk, 2018). In order to show the generality of our approach, we apply our algorithm on two diverse time-series classification problems (Section 4).

2 Background and Notations

We first highlight the properties of certain group transforms by expressing their time-frequency tiling. Then we develop the theoretical tools necessary to build a group transform and illustrate it via the wavelet transform.

2.1 Time-Frequency Tiling

Let's assume that a filter ψ, has narrow localization in time denoted by Δt and a narrow localization in frequency denoted by $\Delta\omega$, then, in the time-frequency plane, these spreads respectively define the width and the height of a rectangular tile, Figure (1) (Mallat, 1999). The area of these tiles, defined by $\Delta t \Delta\omega$, is lower bounded from the Heisenberg uncertainty principle. In other words, the spread of a filter and its Fourier transform are inversely proportional. Following this principle, we can observe that in the case of STFT (resp. WT with a Gabor wavelet), at a given time τ, the signal is transformed by a window of constant bandwidth (resp. proportional bandwidth) modulated by complex exponential resulting in a uniform tiling (resp. proportional) on the frequency axis, Figure (1). In the case of chirp-like filter, as proposed in Baraniuk & Jones (1996), each tile is a sheared rectangular, more generally, an affinely transformed rectangular. In this case, as well, the lower bound area of the sheared rectangular is constrained by the uncertainty principle. As such, the understanding of the benefits of various time-frequency decompositions can be achieved by analyzing how they tile the time-frequency plane. For instance, in the case of WT, the precision in frequency degrades as the frequency increases while its precision in time increases. In the case of STFT, the uniform tiling implies that the precision is constant along the frequency axis. In our propose framework, the LGT allows for an adaptive tiling as illustrated in Figure (1). In the next section we show how the group underlying a group transform induces such a time-frequency tiling.

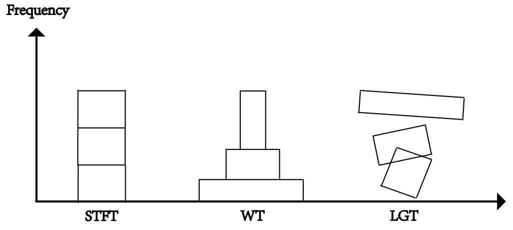

Figure 1: **Time-Frequency Tilings** at a given time τ: (*left*) Short-Time Fourier Transform, i.e., constant bandwidth, (*middle*) Wavelet Transform, i.e., proportional bandwidth, (*right*) Learnable Group Transform, i.e, adaptive bandwidth, the "tiling" is induced by the learned group underlying the filter-bank decomposition.

2.2 Group And Representation

For further details on the group theoretical aspects described in this section, the reader should refer to Vilenkin (1978).

Definition 1. *A group is a set* $\mathbf{G}$ *with a multiplicative operation* $\odot$ *that respects enclosure, identity element, inverse element, and associativity.*

The representation of the group determines its action on a function space and bridges the gap between group theory and linear algebra, allowing to compute the transformation of a function following the rules induced by the specific group at hand. The representation of a group can be thought as a far-reaching generalization of the exponential function property, $\exp(x + y) = \exp(x)\exp(y), \forall x, y \in \mathbb{R}$ (Baraniuk, 1993). In fact, it is defined as,

Definition 2. *A linear continuous representation* ρ *of a group* $\mathbf{G}$ *on the linear space* $\mathbb{H}$ *is defined as*

$$\rho : \mathbf{G} \to GL(\mathbb{H}), \tag{1}$$

where $GL(\mathbb{H})$ *is the the group of linear map in* $\mathbb{H}$ *such that* $\forall g, g^{'} \in \mathbf{G}$

$$\rho(g \odot g') = \rho(g)\rho(g'). \tag{2}$$

It is in fact a homomorphism from the group $\mathbf{G}$ to the group of linear continuous map in $\mathbb{H}$. For instance, let $\mathbb{H}$ be a vector space such as $\mathbb{R}^3$, the representation of the group is induced by 3×3 matrices. In this case, the operation on the right of (2) is a matrix multiplication, where each matrix depends on the group elements g and g'. This concept extends to linear operators acting on functional spaces.

This structure-preserving map defines the action of a group on elements of function spaces. Group transforms such as STFT and CWT can be expressed in such a way by selecting a mother filter space and a group. The representation of the group in the mother filter space provides an operator that takes as input an element of the group and acts on the filter to transform it. A filter-bank can thus be created by iterating this process with different group elements. Therefore, the selected group carries the characteristics of the filter-bank and consequently, the group transform and its time-frequency tiling.

2.3 A Group Transform: The Wavelet Transform

As an introductory example, we consider the creation of a wavelet filter-bank utilizing transformation group. Let's denote by $\mathbf{G}_{\text{aff}}$ the affine group, the so called "$ax + b$" group, where the elements $(a, b) \in \mathbb{R}_+^\star \times \mathbb{R}$, where $\mathbb{R}_+^\star = (0, +\infty)$, where the multiplicative operation of the group $\odot$ is defined by

$$(a, b) \odot (a', b') = (aa', b + ab') \tag{3}$$

Let's define by ρ_{aff} the representation of the affine group in $\mathbb{L}_2(\mathbb{R})$, i.e., $\rho_{\text{aff}} : \mathbf{G}_{\text{aff}} \to GL(\mathbb{L}_2(\mathbb{R}))$, such that ρ_{aff} is a homomorphism as per Definition 2. Its action on square integrable function ψ is defined as

$$[\rho_{\text{aff}}(g)\psi](t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a}), \; t \in \mathbb{R}, \tag{4}$$

where (a, b) are respectively the dilation and translation parameters. The wavelet filter-bank is built by transforming a mother filter, ψ by the representation ρ_{aff} for specific elements of the group. A visualization of this approach for a Morlet wavelet filter can be seen in Figure (3). The wavelet transform of a signal $s_i \in \mathbb{L}_2(\mathbb{R})$ is achieved by

$$\begin{aligned} \mathcal{W}_\psi(g_{(a,b)}, s_i) &= \left\langle \rho_{\text{aff}}(g_{(a,b)})\psi, s_i \right\rangle, \forall g_{(a,b)} \in \mathbf{G}_{\text{aff}}, \quad (5) \\ &= (\rho_{\text{aff}}(g_{(a,0)})\psi \star s_i), \forall g_{(a,0)} \in \mathbf{G}_{\text{aff}}, \quad (6) \end{aligned}$$

where $\langle ., . \rangle$ denotes the inner product, $\star$ the convolution, and $\rho_{\text{aff}}(g_{(a,b)})\psi$ the action of the operator ρ_{aff}, evaluated at the group element $g_{(a,b)}$, on the mother filter ψ as per (4). In practice, the filter-bank is generated by sampling a few elements of the group. For instance, in the case of the dyadic wavelet transform, the dilation parameters follow a geometric progression of common ratio equals to 2. In general, the translation parameter is sampled according to the scaling one (Daubechies, 1992). Notice that in the convolution expression (6), the translation parameter $b = 0$, in fact the convolution operator $\star$ acts as the translation one. In the case where the translation parameter depends on the scaling one, a specific stride is used to perform the discrete convolution.

Note that the STFT can be constructed similarly utilizing the Weyl-Heisenberg group (Feichtinger et al., 2009), whose representation on $\mathbb{L}_2(\mathbb{R})$ consists of frequency modulations and translations. More intricated group representations can be built as in Torrésani (1991) where the combination of the affine group and Weyl-Heisenberg group is considered.

3 Learnable Group Transform

We now develop the generalization of the affine group and its representation on $\mathbb{L}_2(\mathbb{R})$. Its application on signals and learnability is depicted in Figure (2).

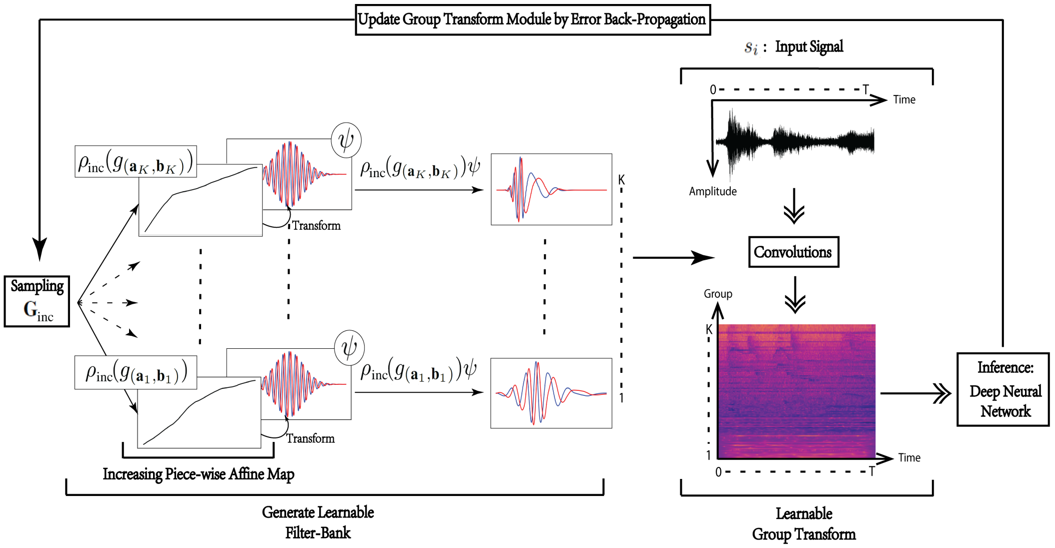
Update Group Transform Module by Error Back-Propagation
s_i : Input Signal
0
T
Time
Amplitude
Convolutions
Group
K
1
0
T
Time
Sampling
$\mathbf{G}_{\text{inc}}$
$\rho_{\text{inc}}(g_{(\mathbf{a}_K,\mathbf{b}_K)})$
ψ
Transform
$\rho_{\text{inc}}(g_{(\mathbf{a}_K,\mathbf{b}_K)})\psi$
K
$\rho_{\text{inc}}(g_{(\mathbf{a}_1,\mathbf{b}_1)})$
ψ
Transform
$\rho_{\text{inc}}(g_{(\mathbf{a}_1,\mathbf{b}_1)})\psi$
1
Increasing Piece-wise Affine Map
Generate Learnable Filter-Bank
Learnable Group Transform
Inference: Deep Neural Network

Figure 2: **Learnable Group Transform:** The first column of the left block is the sampling of the group $\mathbf{G}_{\text{inc}}$ as in Section (3.3) which consists of generating strictly increasing continuous functions $\rho_{\text{inc}}(g_{(\mathbf{a}_k,\mathbf{b}_k)})$ which stands for the representation of the strictly increasing and continuous group for the elements of the group $g_{(\mathbf{a}_k,\mathbf{b}_k)}$, $\forall k \in \{1,\ldots,K\}$, where K denotes the number of filters in the filter-bank. Each generated operators $\rho_{\text{inc}}(g_{(\mathbf{a}_k,\mathbf{b}_k)})$ are applied (curved arrow) to the mother filter denoted by ψ (presently a Morlet wavelet), where the imaginary part is shown in red and the real part in blue. This transformation leads to the filter-bank, $\rho_{\text{inc}}(g_{(\mathbf{a}_k,\mathbf{b}_k)}))\psi$ where $g_{(\mathbf{a}_k,\mathbf{b}_k)} \in \mathbf{G}_{\text{inc}}$. Then, the convolution between this generated filter-bank and the signal leads to the group transform, where the double headed arrows denote the data flow. Each row of the group transform corresponds to the convolution of the signal with each generated filter. The strictly increasing and continuous piece-wise linear functions can be learned efficiently by back-propagating the error induced by the generated group transform.

3.1 The Group of Increasing and Continuous Functions

In order to generalize the classical affine group, we propose the group of strictly increasing and continuous functions. We define its multiplicative operation to be the function composition operation and denote this group by $\mathbf{G}_{\text{inc}} = (C_{\text{inc}}(\mathbb{R}), \odot)$, where

$$C_{\text{inc}}(\mathbb{R}) = \{g \in C(\mathbb{R}) | g \text{ is strictly increasing}\}, \tag{7}$$

and

$$\forall g', g \in C_{\text{inc}}(\mathbb{R}), \;\; g' \odot g = g'(g(t)), \forall t \in \mathbb{R}, \tag{8}$$

where $C(\mathbb{R})$ defines the space of continuous functions defined on $\mathbb{R}$. The identity element of this group is the identity function on $\mathbb{R}$, and the inverse element of g is g^{-1} the inverse of the function g. As this group allows non-linear transformations of the mother filter, the filter-bank derived based on this group has a higher adaptation capability for the pattern of interest in the time-series.

3.2 Representation of the Group

In this work, we are interested in the action of this group on a prototype function. As discussed earlier, the group transform is derived from the representation of the group on the mother filter space. In this section, we present the representation of group $\mathbf{G}_{\text{inc}}$ on square-integrable functions. The transformation of square-integrable prototype functions based on this group allows us to span a wide range of group transforms as illustrated in Figure (1) and to generalize the wavelet transform. Let's define $\rho_{\text{inc}}(g) : \mathbf{G}_{\text{inc}} \to GL(\mathbb{L}_2(\mathbb{R}))$ by

$$[\rho_{\text{inc}}(g)\psi](t) = \psi\big(g(t)\big), \;\; \forall \psi \in \mathbb{L}_2(\mathbb{R}), \forall g \in \mathbf{G}_{\text{inc}}, \tag{9}$$

where ψ denotes the mother filter.

Proposition 1. *ρ_{inc} is a group representation of $\mathbf{G}_{inc}$ on $\mathbb{L}_2(\mathbb{R})$.*

The proof is given in Appendix D.1.

We can see that the increasing and continuous group representation operator ρ_{inc} induces a mapping which depends on the group element $g \in \mathbf{G}_{\text{inc}}$. For instance, if $g = e$, i.e., the identity element of the group, then we have $\rho_{\text{inc}}(e)\psi = \psi$, it is in fact the identity operator in the space of the mother filter. Given a mother filter $\psi \in \mathbb{L}_2(\mathbb{R})$, $\rho_{\text{inc}}(g)\psi, \; \forall g \in \mathbf{G}_{\text{inc}}$ is a transformation of the mother filter with respect to the group element g belonging to the strictly increasing and continuous group which can be visualized in Figure (3). This representation implies a transformation of the mother filter through a time transformation. Note that in signal processing, such a time transformation is called *warping* (Goldenstein & Gomes, 1999; Kerkyacharian et al., 2004).

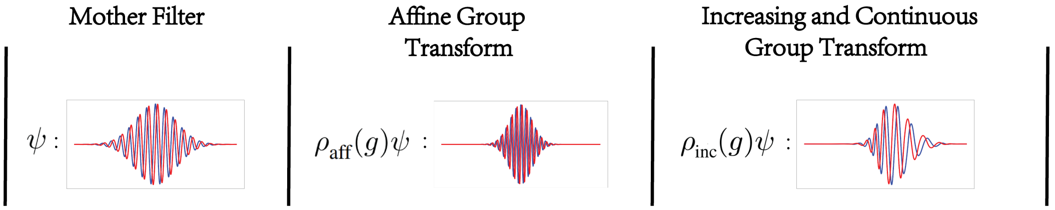

Figure 3: **Transformation of a Morlet Wavelet:** For all the filters, the real part is shown in blue and the imaginary in red. (*left*) Morlet wavelet mother filter. (*middle*) Transformation of the mother filter with respect to the affine group: the parameters of the group are $0 < a < 1$, i.e., contraction, and $b = 0$, i.e., no translation. (*right*) Increasing and continuous group transformation of the mother filter for some randomly generated function $g \in \mathbf{G}_{\text{inc}}$ leading to chirp-like transform.

3.3 Sampling the Group

Sampling the group $\mathbf{G}_{\text{inc}}$ can be achieved by a parametrization of strictly increasing and continuous functions. In the present case, we propose to build a piece-wise affine mapping constrained such that it belongs to the class of strictly increasing and continuous functions. This constrained piece-wise affine mapping is defined as

$$g_{(\mathbf{a},\mathbf{b})}(t) = \sum_{l=1}^{n} (a_l t + b_l)\mathbf{1}_{I_l}(t), \;\; \forall t \in \mathbb{R}, \tag{10}$$

$$\text{s.t.: } \; a_l > 0, \;\; \forall l \in \{1, \dots, n\}, \tag{11}$$

$$b_{l+1} = (a_l - a_{l+1})t_{l+1} + b_l, \;\; \forall l \in \{1, \dots, n-1\}, \tag{12}$$

where $\mathbf{a} = (a_1, \dots, a_n)$, $\mathbf{b} = (b_1, \dots, b_n)$, $\mathbf{1}_{I_l}$ is the indicator function of the intervals $I_l = [t_l, t_{l+1}), \forall l \in \{2, \dots, n-1\}$ and $I_1 = (-\infty, t_1)$, $I_n = [t_n, +\infty)$, and a_l and b_l denote respectively the slope and offset of each piece of the function and n is the number of pieces. As such, for each $(\mathbf{a}, \mathbf{b})$ satisfying the constraints (11) and (12) the function $g_{(\mathbf{a},\mathbf{b})}$ is a sample from the group $\mathbf{G}_{\text{inc}}$.

3.4 Learning the Continuous Piece-wise Affine Increasing Maps

The goal of learning the group transform leads to an optimization problem where the objective is to find samples of the group that will produce the representation that minimizes a specific loss function, thus depending on the signals and the task.

Given a set of signals $\{s_i \in \mathbb{L}_2(\mathbb{R})\}_{i=1}^{N}$ and given a task specific loss function L, we aim at solving the following optimization problem

$$\min_{(\mathbf{a}_1,\mathbf{b}_1)\in\Omega_1,\dots,(\mathbf{a}_K,\mathbf{b}_K)\in\Omega_K} \sum_{i=1}^{N} L\big(F(\mathcal{W}_\psi(\mathbf{g}, s_i))\big), \tag{13}$$

where N denotes the number of signals, K the number of filters, F represents a DNN, $\Omega_k = \{\mathbf{a}_k \in \mathbb{R}_+^n, \mathbf{b}_k \in \mathbb{R}^n | \mathbf{b}_{(k,l+1)} = (\mathbf{a}_{(k,l)} - \mathbf{a}_{(k,l+1)})t_{(l+1,k)} + \mathbf{b}_{(k,l)}\} \; \forall k \in \{1,\dots,K\}$, and $\mathcal{W}_\psi(\mathbf{g}, s_i) = [\mathcal{W}_\psi(g_{(\mathbf{a}_1,\mathbf{b}_1)}, s_i), \dots, \mathcal{W}_\psi(g_{(\mathbf{a}_K,\mathbf{b}_K)}, s_i)]^T$, and $\mathcal{W}_\psi(g_{(\mathbf{a}_k,\mathbf{b}_k)}, s_i) \; \forall k \in \{1,\dots,K\}$ is defined as in (6) without setting the $\mathbf{b}$ parameter to 0. In fact, in the present case, this parameter defines the piece-wise linear maps as opposed to the translation parameter of the affine group.

To solve this optimization problem for diverse time-series, we propose different settings that might be more adapted depending on the type of data and task at hand.

We first propose a normalization of the frequency of the transform filter (denoted in the result tables by nLGT). This normalization helps to reduce the aliasing induced by the filters. We propose to use $\hat{f}$, the normalized frequency f with respect to the maximum slope of the piece-wise affine mapping. For instance, in the case of a Morlet wavelet, the normalization is as follows

$$[\rho_{\text{inc}}(g_{(\mathbf{a},\mathbf{b})})\psi](t) = \pi^{-\frac{1}{4}} \exp\Big(2\pi j \hat{f} g_{(\mathbf{a},\mathbf{b})}(t)\Big) \exp\Big(-\frac{1}{2}(g_{(\mathbf{a},\mathbf{b})}(t)/\sigma)^2\Big),$$

where $\hat{f} = f / \max_{l\in\{1,\dots,n\}} a_l$, j is the imaginary unit, and σ is the width parameter defining the localization of the wavelet in time and frequency. This normalization will be performed for each sample of the group, and thus for each generated filter $k \in \{1,\dots,K\}$ of the filter-bank.

The second setting is a constraint on the domain of the piece-wise affine map as derived in (10) (denoted in the result tables by cLGT). In the following experiments, we propose a dyadic constraint of the domain as in the WT. The support of the filter will thus be close to the support of a wavelet filter-bank. However, the envelop of the filter and the instantaneous frequency will still vary as in the Chirplet Transform (Baraniuk & Jones, 1996).

4 Experiments

For all the experiments and all the settings, i.e., LGT, nLGT, cLGT, cnLGT, the increasing and continuous piece-wise affine map is initialized randomly, and the optimization is performed with Adam Optimizer (Kingma & Ba, 2014), and the number of knots of each piece-wise affine map is 256. The mother filter used for our setting is a Morlet wavelet filter. The code of our LGT framework will be provided on the Github page of the first author.

4.1 Artificial Data: Classification of Chirp Signals

We present an artificial dataset that demonstrates how a specific time-frequency tiling might not be adapted or would require cross-validations for a given task and data. To build the dataset, we generate one high frequency ascending chirp and one descending high-frequency chirp of size 8192 following the chirplet formula provided in (Baraniuk & Jones (1996)). Then for both chirp signals, we add Gaussian noise samples (100 times for each class), see Figures in Appendix (B.1). The task aims at being able to detect whether the chirp is ascending or descending. Both the training and test sets are composed of 50 instances of each class. For all models, set the batch size to 10, the number of epochs to 50. Each experiment was repeated 5 times with randomly sampled train and test set, and the accuracy was the result of the average over these 5 runs. For the case of WT and LGT, the size of the filters is 512. As we can observe in Table (1), the WT, as well as the STFT with few numbers of filters, perform poorly on this dataset. The chirp signals to be analyzed are localized close to the Nyquist frequency, and in the case of WT, we know from Figure 1 that it has a poor frequency resolution in high frequency. Therefore, through time, the small frequency variations of the chirp are not efficiently captured.

Representation + Linear Classifier	*Accuracy*
Wavelet Transform (64 Filters)	53.01 ± 5.1
Short-Time Fourier Transform (64 Filters)	65.1 ± 11.9
Short-Time Fourier Transform (128 Filters)	86.6 ± 9.8
Short-Time Fourier Transform (512 Filters)	**100 ± 0.0**
LGT (64 Filters)	92.9 ± 4.0
nLGT (64 Filters)	95.7 ± 3.3
cLGT (64 Filters)	56.8 ± 1.6
cnLGT (64 Filters)	**100.0 ± 0.0**

Table 1: Testing Accuracy for the Chirp Signals Classification Task

In the case of STFT, if the number of filters is too small, the resolution in frequency is also limited, and thus this variation is as not captured as well. This illustrated the fact in some case neither the proportional-bandwidth nor the constant-bandwidth are suitable. However, using a large window for the STFT increases the frequency resolution of the tiling and thus enables to capture the difference between the two classes. In the case of the LGT, the tiling has adapted to the task and produces good performances except in the cLGT setting. In fact, the domain of the piece-wise linear map is constrained to be dyadic, and thus the adaptivity of the filter bank is reduced which is not suitable for this specific task. For all settings, the visualization of the filters, as well as the representations of the signals, can be found in Appendix (B.1.2,B.1.3).

4.2 Supervised Bird Detection

Representation + Deep Network	*AUC*
MFSC (80 Filters)	77.83 ± 1.34
Conv. Filter init. random (80 Filters)	66.77 ± 1.04
Conv. Filter init. Gabor (80 Filters)	67.67 ± 0.98
Spline Conv. init. random (80 Filters) (Balestriero et al. (2018))	78.17 ± 1.48
Spline Conv. init. Gabor (80 Filters) (Balestriero et al. (2018))	79.32 ± 1.52
LGT (80 Filters)	78.41 ± 1.38
nLGT (80 Filters)	75.50 ± 1.39
cLGT (80 Filters)	79.14 ± 0.83
cnLGT (80 Filters)	**79.68 ± 1.35**

Table 2: Testing AUC for the Bird Detection Task

The proposed Learnable Group Transform is applied to a challenging supervised bird detection task. The dataset is extracted from the Freesound audio archive Stowell & Plumbley (2013). This dataset contains about $7,000$ field recording signals of 10 seconds sampled at 44 kHz, representing slightly less than 20 hours of audio signals. The content of these recordings varies from water sounds to city noises. Among these signals, some contain bird songs that are mixed with different background sounds having more energy than the bird song, see Appendix (B.2.1). The given task is a binary classification where one should predict the presence or absence of bird song. As the dataset is unbalanced, we use the Area Under Curve (AUC) metric. The results we propose for both the benchmarks and our models are evaluated on a test set consisting of 33% of the total dataset. In order to compare with previously used methods, we use the same seeds to sample the train and test set, the batch size, i.e.,10, and the learning rate cross-validation grid as in Balestriero et al. (2018). For each model, the best hyperparameters are selected, and we train and evaluated randomly 10-times the models with early stopping, the results are shown in Table (4.2). While the first layer of the architecture has a model-dependent representation (i.e., MFSC, LGT, Conv. filters,...), we use the state-of-the-art architecture (Grill & Schlüter (2017)) for the DNN architecture, described in Appendix (A.2). Notice that this specific DNN architecture has been designed and optimized for MFSC representation. As we can see, the cnLGT reaches state of the art results.

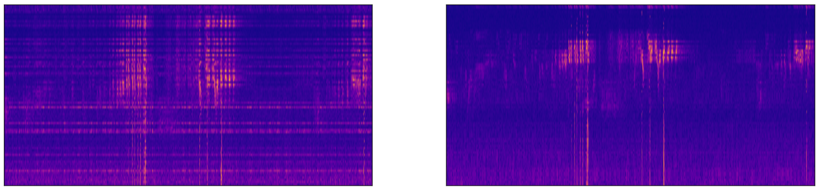

Figure 4: **Learnable Group Transform** - Visualisation of a sample containing a bird song (cLGT), where (*left*) at the initialization and (*right*) after learning. Other representations are displayed in Appendix (B.2.3)

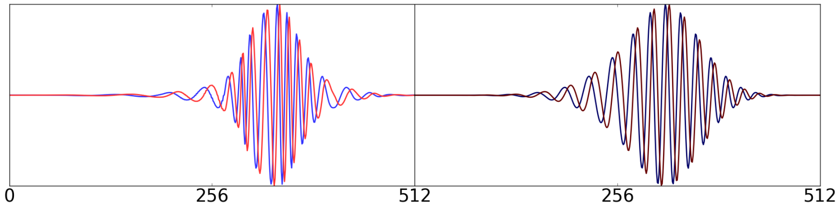

Figure 5: **Learnable Group Transform Filter** - Visualisation of a selected filter (cLGT), where the (*left*) part corresponds to the filter before training and the (*right*) part to the filter after training. The blue and red correspond respectively to the real and imaginary part of the filters. Other filters are displayed in Appendix (B.2.2)

4.3 Haptics Dataset Classification

The Haptics dataset is a classification problem with five classes and 155 training and 308 testing samples from the UCR Time Series Repository Chen et al. (2015), where each time-series has 1092 time samples. As opposed to the bird dataset where features of interests are known, and competitive methods have been established, there are no hand-crafted features that can perform accurately (see Table 3). One can see that our method outperforms other approaches in the cLGT setting while performing the classification with a linear classifier as opposed to other methods using DNN algorithms. This demonstrates the capability of our method to transform the data efficiently while not requiring a further change of basis.

Representation + Classifier	*Accuracy*
DTW (Al-Naymat et al. (2009))	37.7
BOSS (Schäfer (2015))	46.4
Residual NN (Wang et al. (2017))	50.5
COTE (Bagnall et al. (2015))	51.2
Fully Convolutional NN (Wang et al. (2017))	55.1
WD + Convolutional NN (Khan & Yener (2018))	57.5
LGT (96 Filters) + Linear Classifier	53.5
nLGT (96 Filters) + Linear Classifier	50.4
cLGT (96 Filters) + Linear Classifier	**58.2**
cnLGT (96 Filters)+ Linear Classifier	54.3

Table 3: Testing Accuracy for the Haptics Classification Task

5 Conclusion

We proposed to build a novel group transform introducing the group of strictly increasing and continuous functions as well as a tractable way to sample it. From bird detection to haptics classification, our approach competes with state-of-the-art methods without a priori knowledge on the signal power spectrum and outperform classical hand-crafted time-frequency representations. While we have considered only the use of a Morlet mother wavelet, a future approach would be to explore the learnability of the group with the learnability of the mother wavelet proposed in aforecited papers.

REFERENCES

Ghazi Al-Naymat, Sanjay Chawla, and Javid Taheri. Sparsedtw: A novel approach to speed up dynamic time warping. In *Proceedings of the Eighth Australasian Data Mining Conference-Volume 101*, pp. 117–127. Australian Computer Society, Inc., 2009.

Anthony Bagnall, Jason Lines, Jon Hills, and Aaron Bostrom. Time-series classification with cote: the collective of transformation-based ensembles. *IEEE Transactions on Knowledge and Data Engineering*, 27(9):2522–2535, 2015.

Randall Balestriero and Richard Baraniuk. A spline theory of deep networks (extended version). *arXiv preprint arXiv:1805.06576*, 2018.

Randall Balestriero, Romain Cosentino, Hervé Glotin, and Richard Baraniuk. Spline filters for end-to-end deep learning. In *International Conference on Machine Learning*, pp. 373–382, 2018.

Richard Gordon Baraniuk. Shear madness: signal-dependent and metaplectic time-frequency representations. 1993.

Richard Gordon Baraniuk and Douglas L Jones. Wigner-based formulation of the chirplet transform. *IEEE Transactions on signal processing*, 44(12):3129–3135, 1996.

Emre Cakir, Ezgi Can Ozan, and Tuomas Virtanen. Filterbank learning for deep neural network based polyphonic sound event detection. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pp. 3399–3406. IEEE, 2016.

Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive, July 2015. `www.cs.ucr.edu/~eamonn/time_series_data/`.

Ronald R Coifman and M Victor Wickerhauser. Entropy-based algorithms for best basis selection. *Information Theory, IEEE Transactions on*, 38(2):713–718, 1992.

Romain Cosentino, Randall Balestriero, and Behnaam Aazhang. Best basis selection using sparsity driven multi-family wavelet transform. In *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 252–256, Dec 2016. doi: 10.1109/GlobalSIP.2016.7905842.

Romain Cosentino, Randall Balestriero, Richard Baraniuk, and Ankit Patel. Overcomplete frame thresholding for acoustic scene analysis. *arXiv preprint arXiv:1712.09117*, 2017.

Ingrid Daubechies. *Ten lectures on wavelets*, volume 61. Siam, 1992.

Hans G Feichtinger and KH Gröchenig. Banach spaces related to integrable group representations and their atomic decompositions, i. *Journal of Functional analysis*, 86(2):307–340, 1989.

Hans G Feichtinger, Werner Kozek, and Franz Luef. Gabor analysis over finite abelian groups. *Applied and Computational Harmonic Analysis*, 26(2):230–248, 2009.

Siome Goldenstein and Jonas Gomes. Time warping of audio signals. In *cgi*, pp. 52. IEEE, 1999.

Rémi Gribonval and Emmanuel Bacry. Harmonic decomposition of audio signals with matching pursuit. *IEEE Transactions on Signal Processing*, 51(1):101–111, 2003.

Thomas Grill and Jan Schlüter. Two convolutional neural networks for bird detection in audio signals. In *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, Kos Island, Greece, August 2017. URL `http://ofai.at/~jan.schlueter/pubs/2017_eusipco.pdf`.

Alex Grossmann, Jean Morlet, and Thierry Paul. Transforms associated to square integrable group representations. i. general results. *Journal of Mathematical Physics*, 26(10):2473–2479, 1985.

Alex Grossmann, Jean Morlet, and Thierry Paul. Transforms associated to square integrable group representations. ii: examples. In *Annales de l'IHP Physique théorique*, volume 45, pp. 293–309, 1986.

Gérard Kerkyacharian, Dominique Picard, et al. Regression in random design and warped wavelets. *Bernoulli*, 10(6):1053–1105, 2004.

Haidar Khan and Bulent Yener. Learning filter widths of spectral decompositions with wavelets. In *Advances in Neural Information Processing Systems*, pp. 4601–4612, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Milan Korda and Igor Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, 2018.

Fabien Lelandais and Herve Glotin. Mallat's matching pursuit of sperm whale clicks in real-time using daubechies 15 wavelets. In *New Trends for Environmental Monitoring Using Passive Systems, 2008*, pp. 1–5. IEEE, 2008.

Yang Liu and Sergey Fomel. Seismic data analysis using local time-frequency decomposition. *Geophysical Prospecting*, 61(3):516–525, 2013.

Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

Stéphane G Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, 1993.

Mirco Ravanelli and Yoshua Bengio. Interpretable convolutional filters with sincnet. *arXiv preprint arXiv:1811.09725*, 2018.

C Saritha, V Sukanya, and Y Narasimha Murthy. Ecg signal analysis using wavelet transforms. *Bulg. J. Phys*, 35(1):68–77, 2008.

Patrick Schäfer. The boss is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530, 2015.

Léonard Seydoux, Nikolaï M Shapiro, Julien de Rosny, Florent Brenguier, and Matthieu Landès. Detecting seismic activity with a covariance matrix analysis of data recorded on seismic arrays. *Geophysical Journal International*, 204(3):1430–1442, 2016.

Dan Stowell and Mark D. Plumbley. An open dataset for research on audio field recording archives: freefield1010. *CoRR*, abs/1309.5275, 2013. URL `http://arxiv.org/abs/1309.5275`.

Bruno Torrésani. Wavelets associated with representations of the affine weyl–heisenberg group. *Journal of Mathematical Physics*, 32(5):1273–1279, 1991.

Naum Yakovlevich Vilenkin. *Special functions and the theory of group representations*, volume 22. American Mathematical Soc., 1978.

Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 international joint conference on neural networks (IJCNN)*, pp. 1578–1585. IEEE, 2017.

David A Winkler and Tu C Le. Performance of deep and shallow neural networks, the universal approximation theorem, activity cliffs, and qsar. *Molecular informatics*, 36(1-2):1600118, 2017.

Neil Zeghidour, Nicolas Usunier, Iasonas Kokkinos, Thomas Schaiz, Gabriel Synnaeve, and Emmanuel Dupoux. Learning filterbanks from raw speech for phone recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5509–5513. IEEE, 2018.

A Architecture Details

A.1 Artificial Data

Group Transform + Complex Modulus + Log
Dense Layer (1 sigmoid)

After the Group Transform, a batch-normalization is applied.

A.2 Supervised Bird Detection

Group Transform + Complex Modulus + Log + Average-Pooling (stride:$(1, 512)$ size:$(1, 1024)$)
Conv2D. layer (16 filters 3×3) and Max-Pooling (3×3) and ReLU
Conv2D. layer (16 filters 3×3) and Max-Pooling (3×3) and ReLU
Conv2D. layer (16 filters 3×1) and Max-Pooling (3×1) and ReLU
Conv2D. layer (16 filters 3×1) and Max-Pooling (3×1) and ReLU
Dense layer (256) and ReLU
Dense layer (32) and ReLU
Dense layer (1 sigmoid)

At each layer a batch-normalization is applied and for the last three layers a 50% dropout is applied as in (Grill & Schlüter (2017)). The dimension of the input of the DNN presented is the same for the different benchmarks.

A.3 Haptics Data

Group Transform + Complex Modulus + Log + Average-Pooling (stride:$(1, 64)$ size:$(1, 128)$)
Dense Layer (5 softmax)

After the Group Transform, a batch-normalization is applied.

B Additional Figures

B.1 Artificial Data

B.1.1 Data

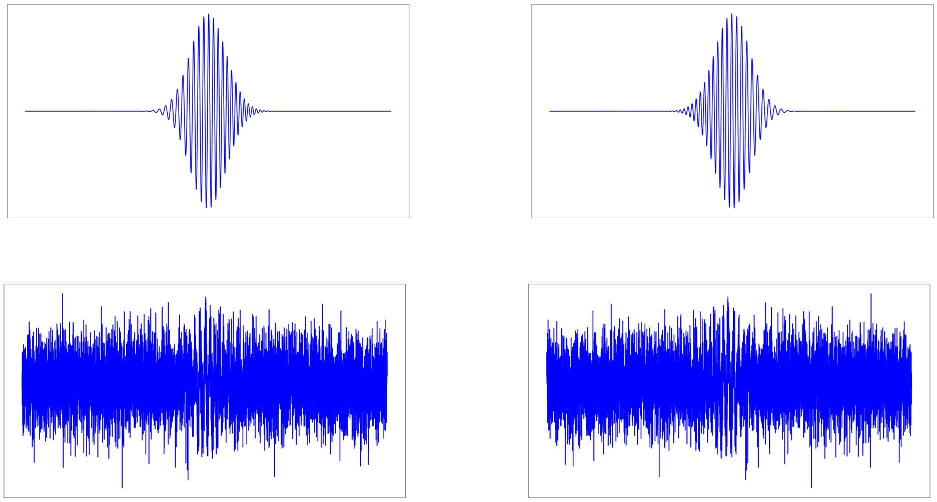

Figure 6: **Artificial Dataset**: (*Top Left*) Ascending Chirp, (*Top Right*) Descending Chirp, i.e. class 0, (*Bottom Left*) Ascending Chirp plus Gaussian noise, (*Bottom Right*) Descending Chirp plus Gaussian noise, i.e., class 1. The samples contained in the training and testing set are higher in frequency and close to the Nyquist frequency.

B.1.2 Filters

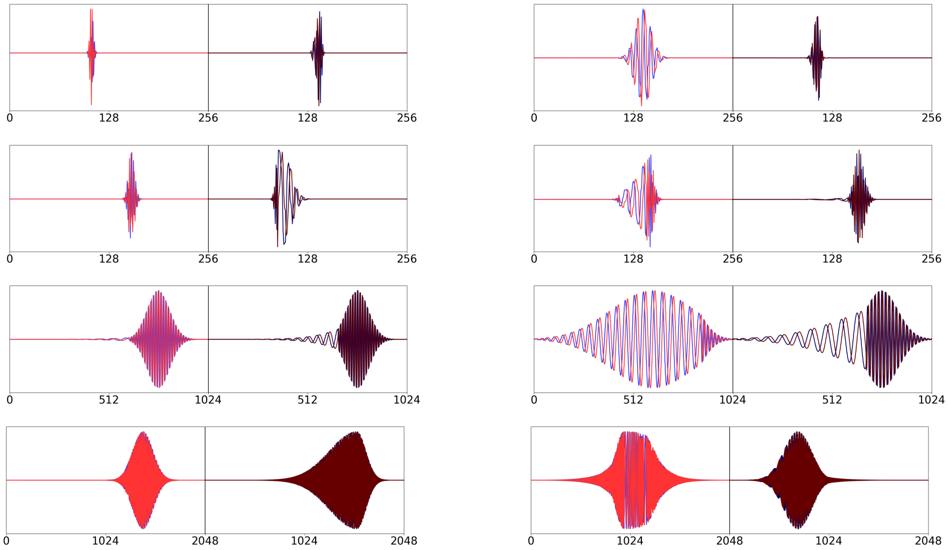

Figure 7: **Learnable Group Transform Filters** for the Artificial Data - Each row displays two selected filters (left and right sub-figure) for different settings: (*from top to bottom*) LGT, nLGT, cLGT, cnLGT. For each subfigure, the left part corresponds to the filter before training and the right part to the filter after training. The blue and red denotes respectively the real and imaginary part of the filters.

B.1.3 Group Transform

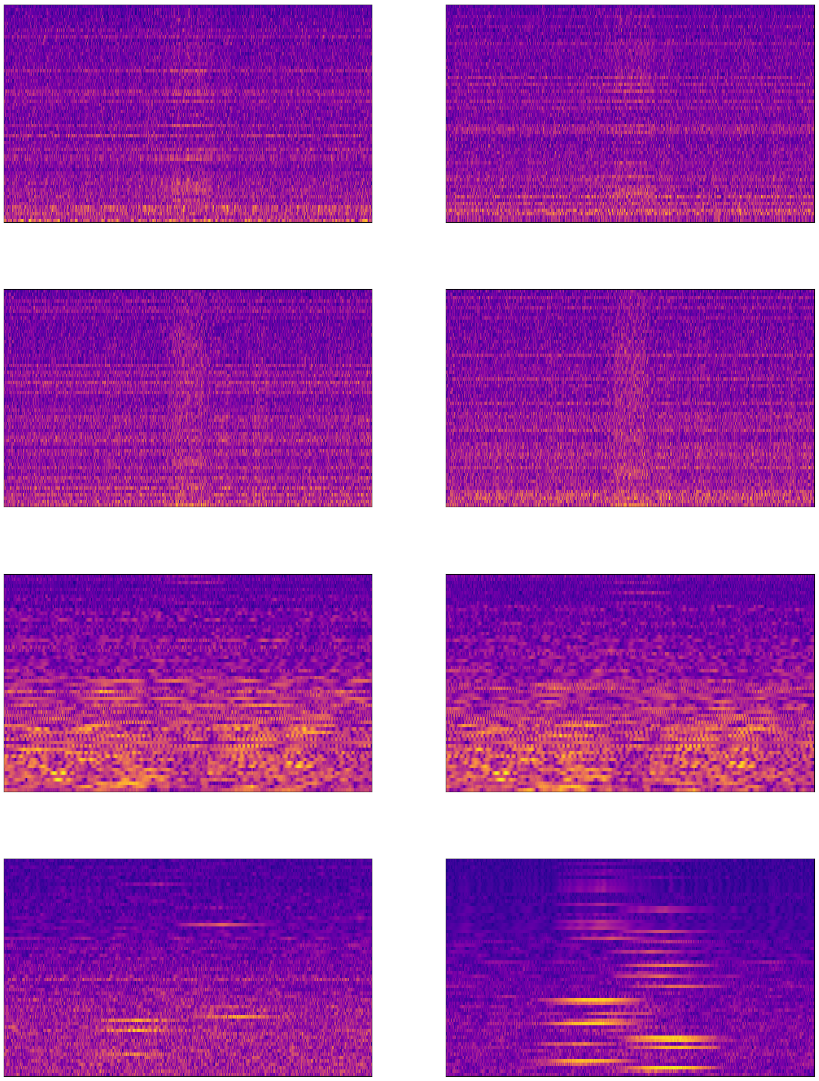

Figure 8: **Learnable Group Transform** - Visualisation of an ascending chirp sample, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

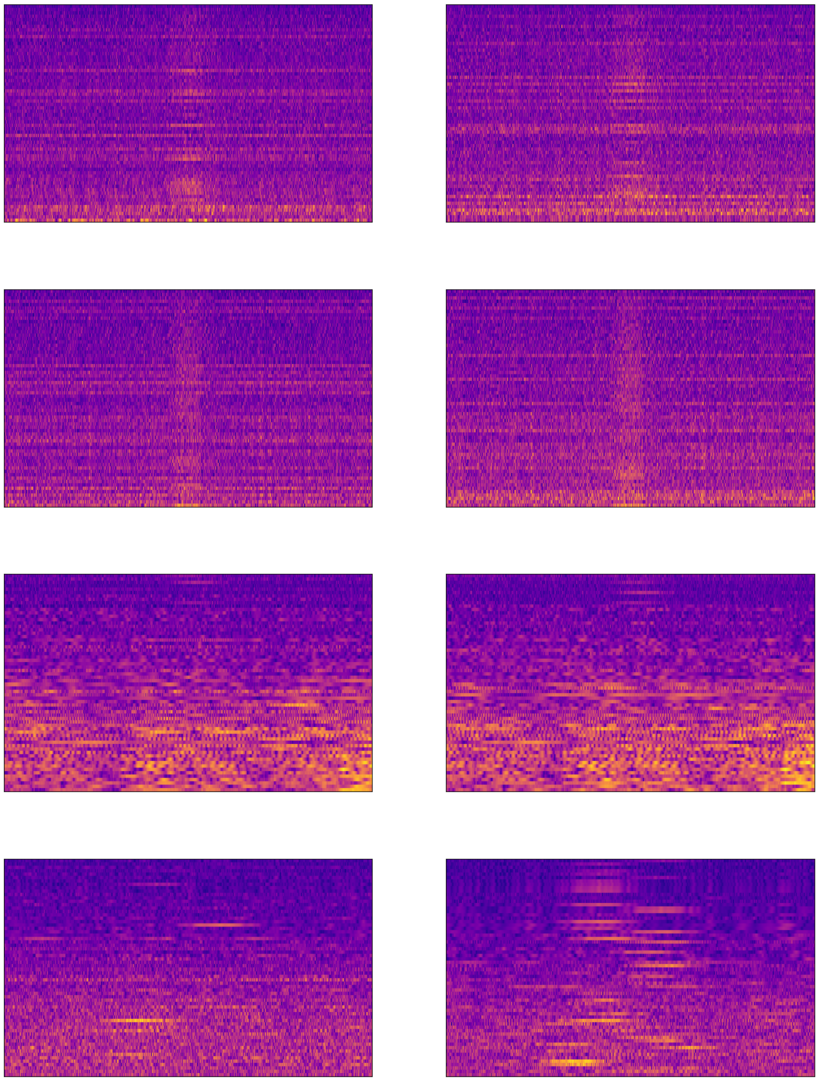

Figure 9: **Learnable Group Transform** - Visualisation of a descending chirp sample, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

B.2 Supervised Bird Detection

B.2.1 Data

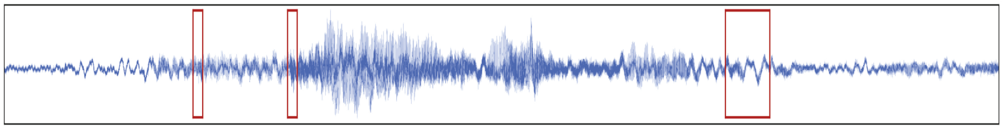

Figure 10: **Bird Detection Dataset** - Sample containing a bird song. The red boxes are the locations of the bird song.

Each data sample, normalized, centered and subsampled by two before experiment.

B.2.2 Filters

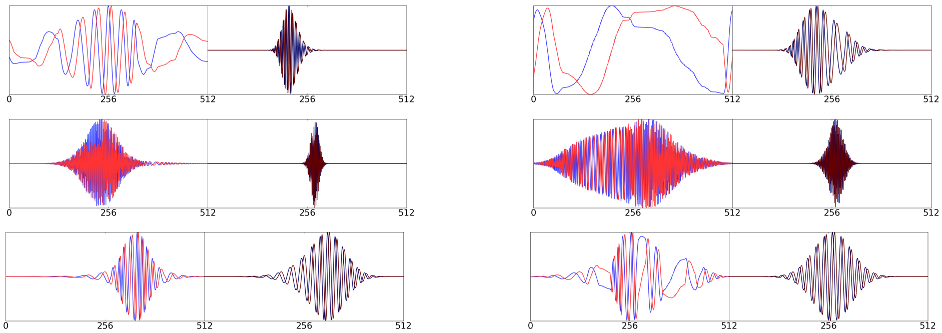

Figure 11: **Learnable Group Transform Filters** for the Bird Detection Data - Each row displays two selected filters (left and right sub-figure) for different settings: (*from top to bottom*) LGT, nLGT, cLGT. For each subfigure, the left part corresponds to the filter before training and the right part to the filter after training.

B.2.3 Group Transform

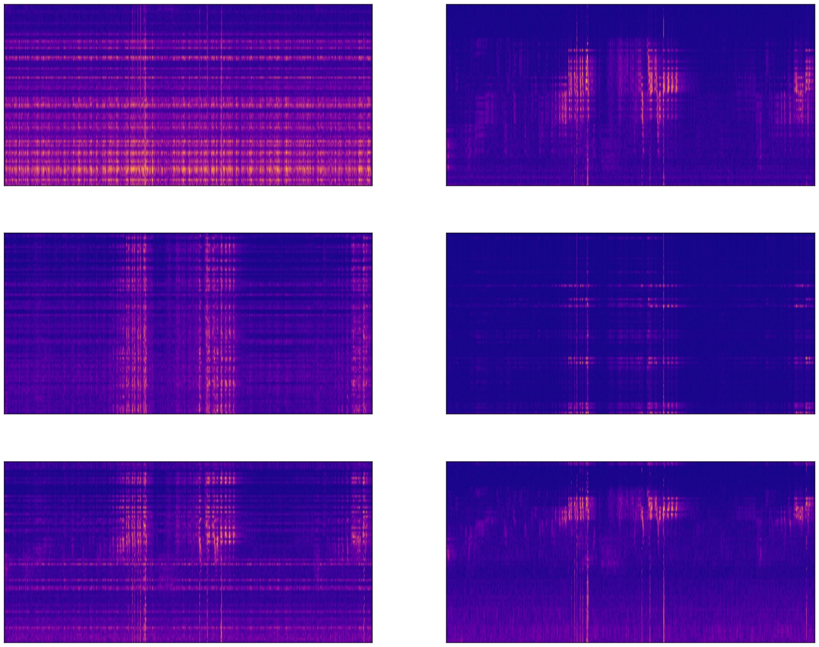

Figure 12: **Learnable Group Transform** - Visualisation of a sample containing a bird song, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT.

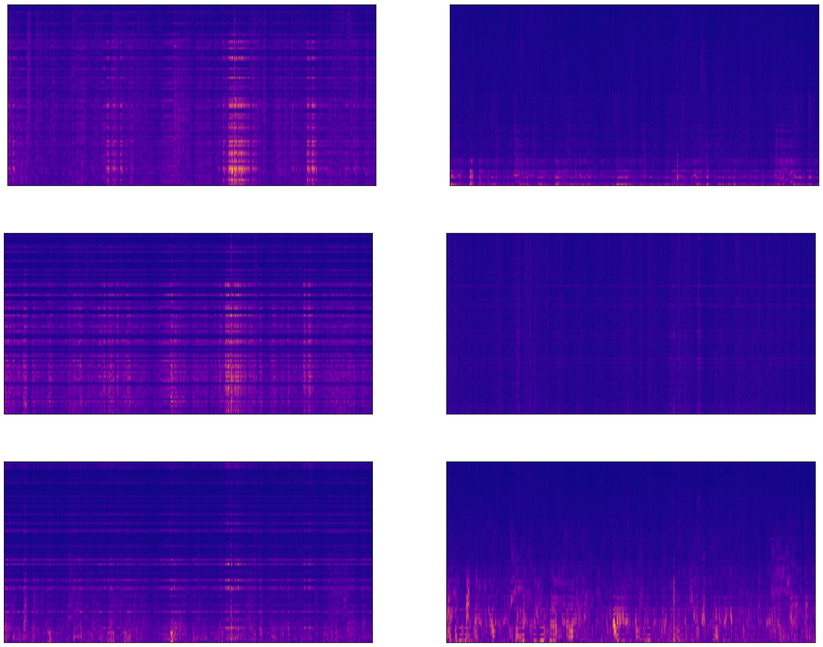

Figure 13: **Learnable Group Transform** - Visualisation of a sample without a bird song, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT.

B.3 Haptics Data

B.3.1 Data

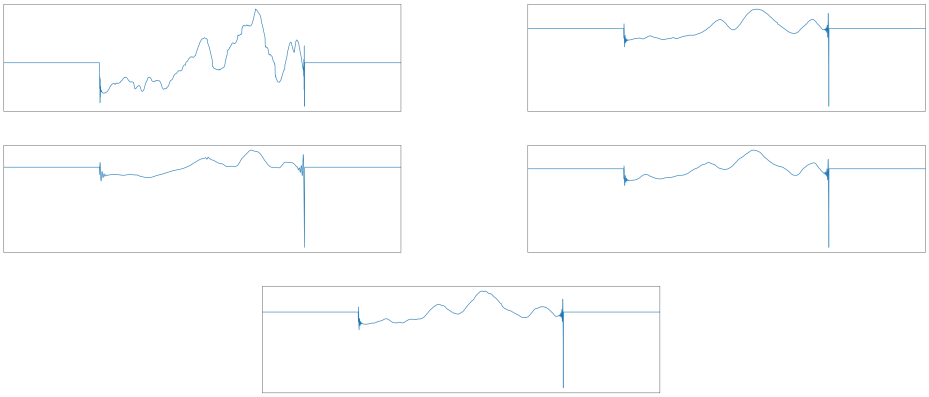

Figure 14: **Haptic Dataset** - Sample of each class of the Haptic dataset.

Each data is centered and normalized. For the experiments, the number of epochs is set to 1000 and we perform early-stopping and obtain the testing accuracy at this specific epoch as in Khan & Yener (2018), the batch size was set to 64. In order to avoid overfitting, we perform different asymmetric zeros-paddings on the training samples. For the testing samples, we perform a symmetric zeros-padding (512 zeros on each side of the signals).

B.3.2 Filters

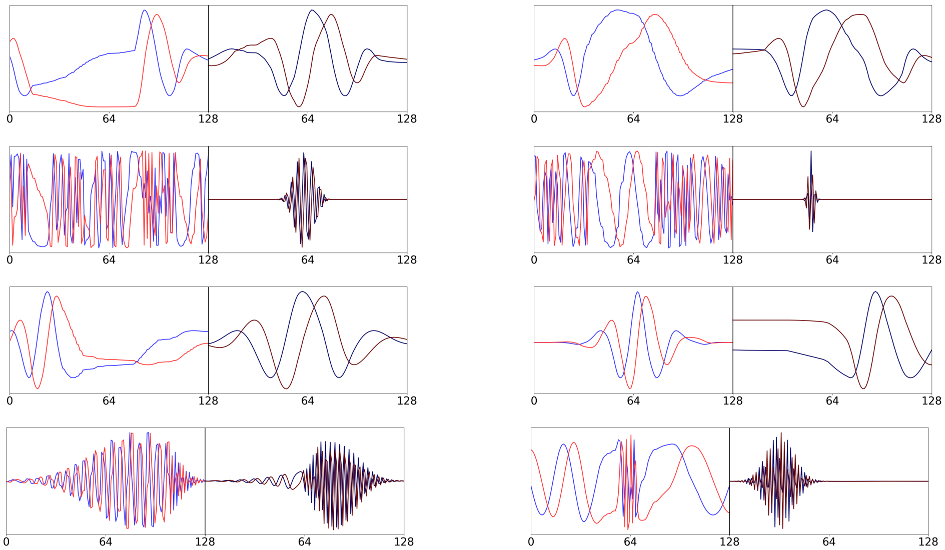

Figure 15: **Learnable Group Transform Filters** for the Haptics Data - Each row displays two selected filters (left and right sub-figure) for different settings: (*from top to bottom*) LGT, nLGT, cLGT, cnLGT. For each subfigure, the left part corresponds to the filter before training and the right part to the filter after training. The blue and red denotes respectively the real and imaginary part of the filters.

B.3.3 Group Transform

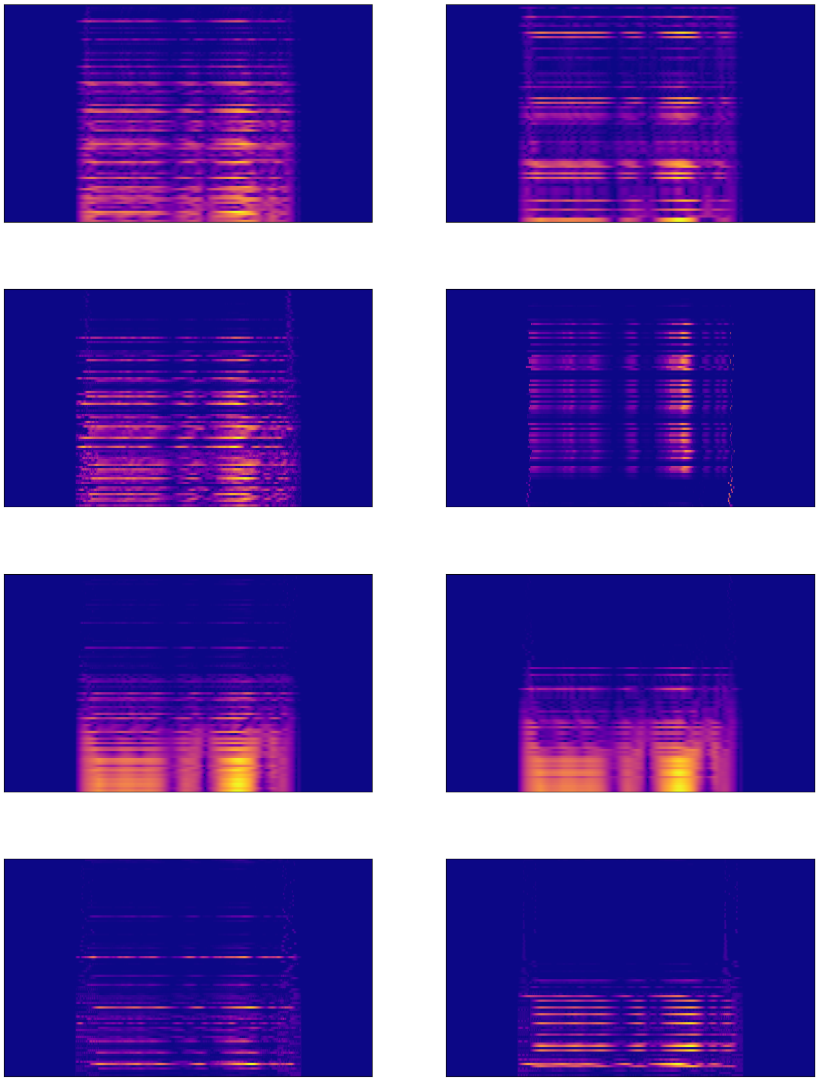

Figure 16: **Learnable Group Transform** - Visualisation of a sample belonging to class 1, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

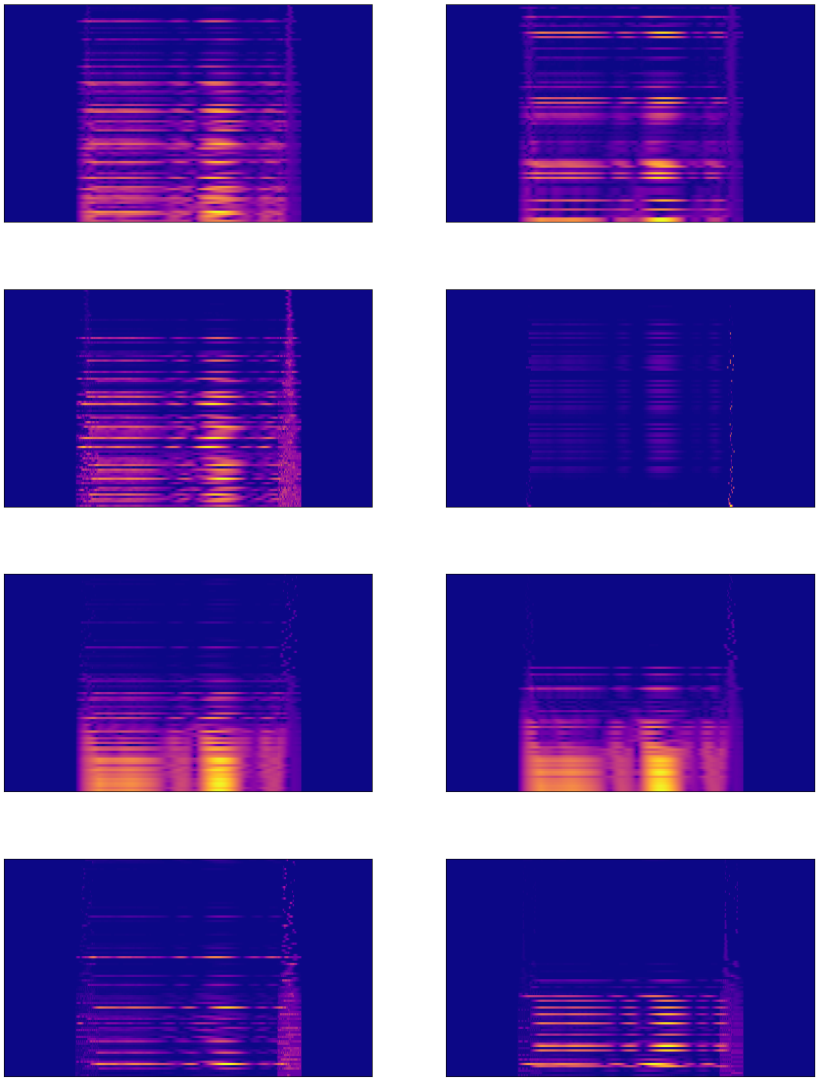

Figure 17: **Learnable Group Transform** - Visualisation of sample belonging to class 2, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

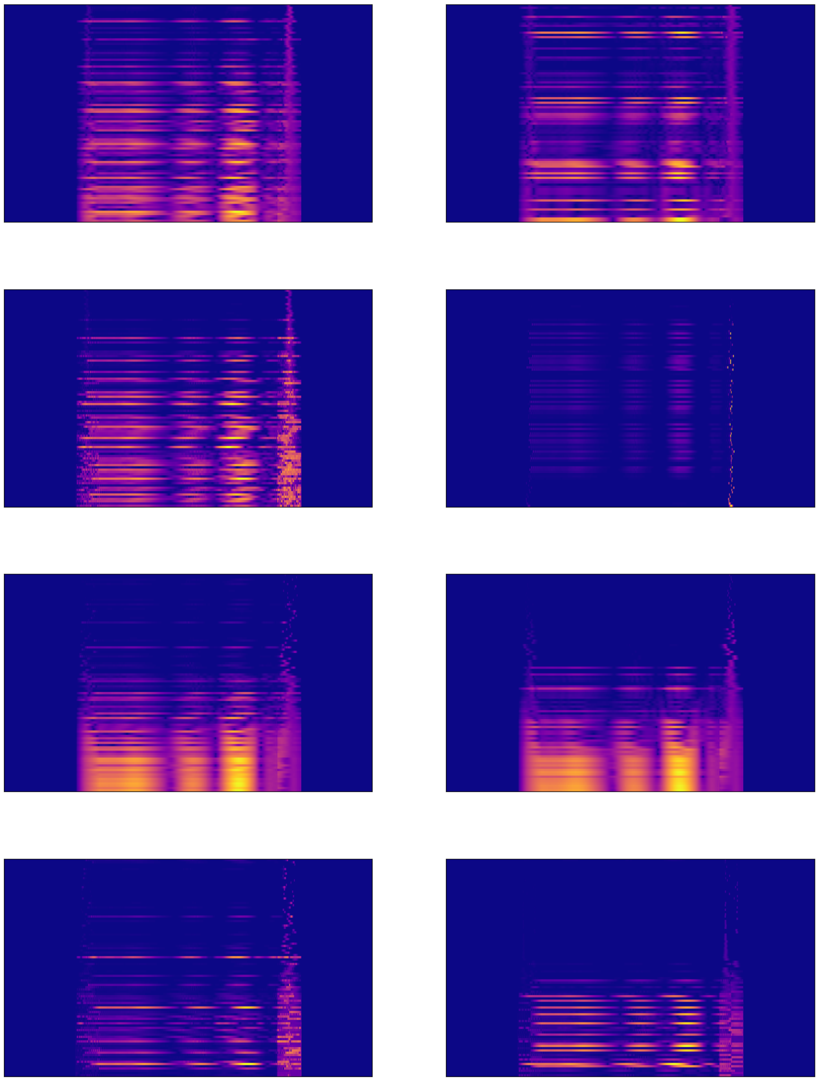

Figure 18: **Learnable Group Transform** - Visualisation of sample belonging to class 3, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

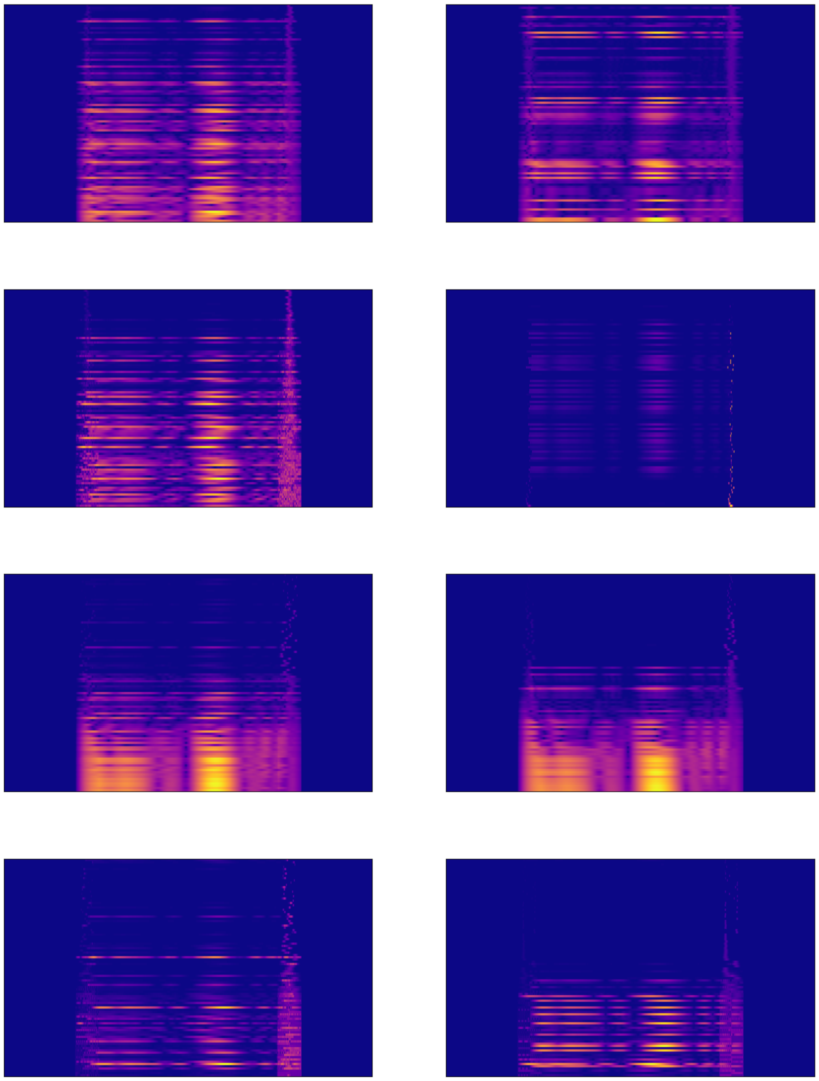

Figure 19: **Learnable Group Transform** - Visualisation of a sample belonging to class 4, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

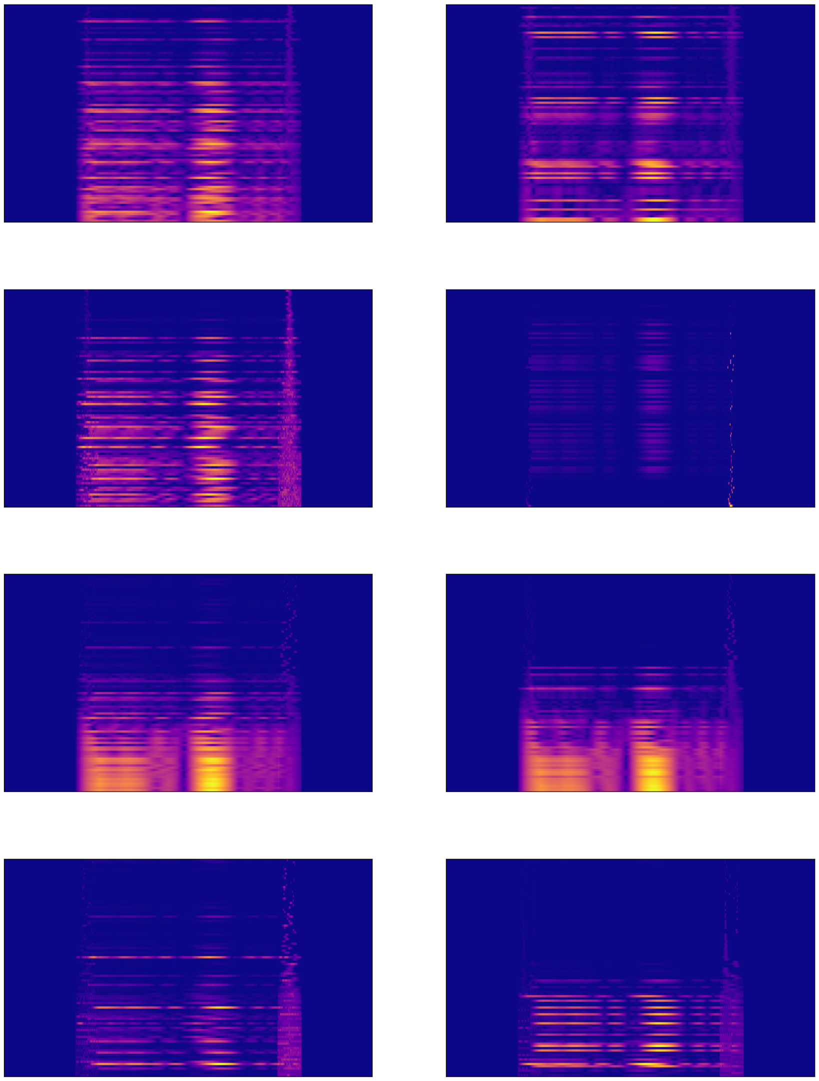

Figure 20: **Learnable Group Transform** - Visualisation of a sample belonging to class 5, where for each row (*left*) at the initialization and (*right*) after learning. Each row displays a different setting: (*from top to bottom*): LGT, nLGT, cLGT, cnLGT.

C Group Parameter Optimization

In order to learn the group transform module, we can use the back-propagation algorithm and a gradient-based optimization technique such that the parameters of the group transform module, denoted by g, can be learned jointly with the parameters of the DNN, or any other differentiable algorithm taking as input the learnable time-frequency representation. Using the notations of Section 3.4 where L denotes a loss function and F a DNN, the learnability of the optimal group transform leading to the most suitable time-frequency representation is performed by the chain rule,

$$\frac{\partial L}{\partial g_k} = \frac{\partial L}{\partial[F(\mathcal{W}_\psi(g_k, s_i))]} \times \frac{\partial[F(\mathcal{W}_\psi(g_k, s_i))]}{\partial g_k}, \forall i \in \{1, \ldots, N\}, \ \forall k \in \{1, \ldots, K\}, \ g_k \in \mathbf{G}_{\text{inc}},$$

where $[\mathcal{W}_\psi(g_k, s_i)]$ is the convolution of the signals s_i with the transformed filter $\rho_{\text{inc}}(g_k)\psi$ as defined in (6).

D PROOFS

D.1 PROOF THEOREM 1

Proof. Let $g, g' \in \mathbf{G}_{\text{inc}}$, then

$$\begin{aligned}[\rho_{\text{inc}}(g' \circledast g)\psi](t) &= \psi((g' \circledast g)(t)) \\ &= \psi(g'(g(t)))\end{aligned}$$

and,

$$\begin{aligned}[\rho_{\text{inc}}(g')\rho_{\text{inc}}(g)\psi](t) &= [\rho_{\text{inc}}(g')\psi](g(t)) \\ &= \psi(g'(g(t)))\end{aligned}$$

which verifies the homogeneity property. The linearity is implied by,

$$[\rho_{\text{inc}}(g)(\kappa\psi_1 + \psi_2)](t) = (\kappa\psi_1 + \psi_2)(g(t)) = \kappa\psi_1(g(t)) + \psi_2(g(t)), \forall t \in \mathbb{R}.$$

where $\psi_1, \psi_2 \in \mathbb{L}_2(\mathbb{R})$ and $\kappa \in \mathbb{R}$. It is in fact a Koopman operator Korda & Mezić (2018). □