Separable Wavelet Scattering

M.W. Rademan, D.J.J Versfeld, J.A. du Preez

***Abstract*—Wavelet scattering is a widely used feature extraction method due to it efficacy in extracting invariant features, while retaining any lost high frequency information resulting from averaging to obtain the desired amount of invariance. By generalising a 1-dimensional scattering transform, we extend its definition to an arbitrary number of independent dimensions. We show that, in a modern deep learning setting, separable wavelet scattering performs similarly to its non-separable counterparts with the MNIST hand-written digit dataset. We additionally demonstrate state-of-the-art results for a subset of the MedMNIST3D datasets.**

***Index Terms*—classification, MNIST, MedMNIST, wavelet scattering**

I. Introduction

Wavelet scattering (WS) has seen widespread use in classification applications as a powerful feature extraction method. It is an especially effective method for small datasets, since the feature filters are reminiscent of a convolutional neural network (CNN) front-end, while requiring no learning. The resulting features are invariant across all averaged dimensions, and exhibit separable class subspaces, allowing linear classifiers to be used with high efficacy [1].

Various forms of WS exist, with 1-dimensional (1D) scattering first proposed by Anden and Mallat [2], [**?**], which was later extended to 2 [1] and 3 dimensions [3], [4]. Additional and more specialised forms for WS include rotation-invariant scattering [5] for the classification of textures and joint-time-frequency (TF) scattering [**?**], [6]. Joint-TF scattering utilises a separable 2-dimensional (2D) filter that operates on the first level scattering scalogram, denoted by the operator $\mathcal{U}_1$, which is the only apparent usage of separable filters in the current scattering literature.

In a deep-learning setting, sensible initialisation of filters prior to learning can significantly improve performance and interpretability [7]. The greatest advantage of separable filters is their computation speed when computing convolutions directly, as is performed in CNNs [8]. Additionally, separable filters reduce the number of parameters of a CNN if the filters are configured to be learnable. Learnable filters can typically improve performance compared to their fixed-filters counterpart [**?**].

At first glance, a WS decomposition seems to be prohibitively expensive, but fast algorithms are possible due to the limited bandwidth of filter output. Fast algorithms utilise downsampling to take advantage of the demodulation of band-limited signals that result from the convolution of analytic wavelets [1], [3], [2]. Path pruning is also used to ignore filter combinations which have negligible energy. These algorithms are the standard implementation on many platforms, such as the Kymatio python package [9] and MATLAB [10].

The Morlet wavelet is the most widely used wavelet filter for a scattering filter bank implementation. Since the Morlet does not have compact support, implementations vary when considering the bandwidth/time support of the Morlet. This affects how Morlet finite impulse response (FIR) filters are discretised and truncated, how scattering paths with negligible energy are pruned, and how filters overlap in the frequency domain.

In this work, we generalise 1D scattering to an arbitrary number of dimensions, which we refer to as the separable scattering transform. We propose flexible definitions for the Morlet bandwidth and filter overlap, which allows for an exact definition of near-optimal convolution computations with downsampling. Our implementation of separable scattering is graphics processing unit (GPU) accelerated, similar to the Kymatio implementations [9].

We demonstrate using the MNIST handwritten digit dataset [11] that separable wavelet filters still perform adequately in a neural network (NN) classification setting compared to the conventional 2D scattering filters. We demonstrate the arbitrarily scalable dimensionality of the separable scattering transform with three-dimensional medical datasets from MedMNIST3D [12]. Separable scattering coefficients show state-of-the-art (SOTA) results for some of the MedMNIST3D datasets when combined with a simple NN, while performing comparably to CNN benchmarks on datasets which do not indicate SOTA results.

II. Separable Morlet Filterbank

A. Morlet

We define a 1D zero-mean Gaussian as

$$\theta_{\sigma_t}(t) = \frac{1}{\sqrt{2\pi\sigma_t^2}} e^{-\frac{1}{2}\left(\frac{t}{\sigma_t}\right)^2}, \tag{1}$$

which has a Fourier transform transform $\theta_{\sigma_t}(t) \overset{\mathcal{F}}{\longleftrightarrow} \hat{\theta}_{\sigma_\omega}(\omega)$, where $\sigma_\omega = \frac{1}{\sigma_t}$.

For an arbitrary bandwidth definition, we define the bandwidth-to-σ ratio $\beta \in \mathbb{R}^+$, such that the one-sided bandwidth of a zero-mean Gaussian is $\beta\sigma_\omega$. For reference, Kymatio's implementation of 2D scattering indirectly defines $\beta \approx 2.5$ [9].

A 1D Morlet ψ has energy concentrated around 1 rad/s, with most of its energy contained in the interval $\omega \in [1-\frac{1}{Q}, 1+\frac{1}{Q}]$. $Q \in \mathbb{R}^+$ is defined as the number of wavelets per octave. The mother wavelet is given as

$$\psi(t) = \theta_{\sigma_t}(t)\left(e^{jt} - \theta_0\right), \tag{2}$$

where $\theta_0 = \frac{\theta_{\sigma_t}(-1)}{\theta_{\sigma_t}(0)}$ to ensure zero mean: $\hat{\psi}(0) = 0$.

For reasons that will become apparent in section II-B, we defined a 1D-wavelet dilated by a factor λ as

$$\psi_\lambda(t) = \begin{cases} \psi(\lambda t), & \lambda \neq 0 \\ \phi_j(t), & \lambda = 0 \end{cases}, \tag{3}$$

where ϕ represents the 1D low-pass filter (LPF) utilised in scattering computations for the current dimension (j).

A n-dimensional separable wavelet may be constructed similarly, with

$$\psi_{\boldsymbol{\lambda}}(\mathbf{u}) = \prod_{i=0}^{n-1} \psi_{\lambda_i}(u_i), \tag{4}$$

where $\mathbf{u} = (u_1, ..., u_n)^T$ is the vector containing the dimensions of interest.

A filterbank is constructed by dilating the mother wavelet with a set of dilation factors $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_n)^T \in \Lambda_1 \times ... \times \Lambda_n$, with Λ_i the set of dilation factors for the variable u_i. The dilation factor λ_i is also the centre frequency of the dilated wavelet in rad/s.

B. 1D Filterbank Construction

Suppose a 1D Morlet has a centre frequency λ_0. The following filter is placed at $\lambda_0 2^{\frac{1}{Q}}$. To define the amount of overlap between filters, it is useful express the corresponding filter's frequency standard deviation (SD) σ_ω in terms of the distance between filters in the frequency domain. If the wavelet at λ_0 requires a decay equivalent to $\alpha\sigma_\omega$ SDs at the next wavelet at $\lambda_0 2^{\frac{1}{Q}}$. It follows that $\sigma_\omega = \frac{1}{\sigma_t} = \frac{1}{\alpha}\left(2^{\frac{1}{Q}} - 1\right)$.

We refer to α as the clearance factor. More overlap will result in more redundancy in the frequency representation, but also shorter filter impulse responses.

The provided definitions of filter clearance (α) and bandwidth (β) may also be extended to non-separable wavelets, such as the rotational construction employed in [1]. However, a conversion is required to find the bandwidth in each axis for a specific wavelet for rotationally constructed filterbanks, in order to employ the downsampling strategies proposed in this paper.

All filterbank descriptions in this paper utilise normalised sampling frequency, i.e., the sample frequency is $f_s = 1$.

For some audio applications, for example speech recognition, low-frequency content is negligible, motivating filterbank construction starting at a specified frequency ω_0. Otherwise, the entire frequency domain may be covered by setting ω_0 to start at a position specified by ϕ and α.

The LPF ϕ is constant across all layers in a 1D scattering transform, and is chosen as $\phi(t) = \theta_{\sigma_{\phi,t}}(t)$, where $\sigma_{\phi,t} = \dfrac{d}{\pi\beta}$.

The scattering transform requires that the time support of all filters do not exceed the time support of ϕ, ensuring no filters contain time information exceeding the invariance scale d. d is equivalently defined as the total downsample factor of the scattering tranform, and is unique for each dimension. Equivalently, the frequency SD of the filters may not exceed the SD of $\phi(\omega)$. For brevity, we denote the frequency SDs as $\sigma_{\lambda,\omega}$ and $\sigma_{\phi,\omega}$ for the Morlets and the LPF respectively.

Since the maximum time support (minimum bandwidth) is a function of d, all dilated wavelets have their Gaussian envelopes restricted to a maximum time support of $\sigma_{\phi,t}$. This requires linearly spaced filters until the dilated bandwidth is larger than the LPF bandwidth. To construct a set of positive dilation factors $\lambda \in \Lambda^+$, refer to algorithm 1.

Algorithm 1 Λ^+ construction.

$\sigma_{\phi,\omega} \leftarrow \dfrac{\pi\beta}{d}$
if frequency limited **then**
Ensure: $\omega_0 \geq \dfrac{\pi\alpha}{d}$
$\lambda \leftarrow \omega_0$
else
$\lambda \leftarrow \dfrac{\pi\alpha}{d}$
end if
$\Lambda^+ \leftarrow \varnothing$
$\sigma_\omega \leftarrow \dfrac{1}{\alpha}\left(2^{\frac{1}{Q}} - 1\right)$
$\sigma_{\lambda,\omega} \leftarrow \lambda\sigma_\omega$
while $\sigma_{\phi,\omega} > \sigma_{\lambda,\omega}$ and $\lambda < \pi$ **do**
$\Lambda^+ \leftarrow \Lambda^+ \cup \{\lambda\}$
$\lambda \leftarrow \lambda + \alpha\sigma_{\phi,\omega}$
$\sigma_{\lambda,\omega} \leftarrow \lambda\sigma_\omega$
end while
while $\lambda < \pi$ **do**
$\Lambda^+ \leftarrow \Lambda^+ \cup \{\lambda\}$
$\lambda \leftarrow \lambda 2^{\frac{1}{Q}}$
$\sigma_{\lambda,\omega} \leftarrow \lambda\sigma_\omega$
end while

Only positive λ's have been defined thus far, which provides inadequate coverage of the frequency domain in multiple dimensions. For real input signals, it is only necessary to cover half of one of the dimensions (only positive λ's), whereas full coverage (both negative and positive λ's) is required for additional dimensions. On-axis coverage is also required, in which each ψ_λ must be multiplied with a Gaussian (zero-frequency wavelet), which the dilated wavelet definition in equation (3) defines as $\lambda = 0$. A similar construction procedure is followed in [6].

Given m 1D filterbanks, with $m \geq 2$, each having Morlet filters with a positive set of lambdas Λ_i^+ and invariance scales $\mathbf{d} \in \mathbb{N}^m$, with $i = 1, ..., m$ indexing the dimension, we construct the m-dimensional filterbank with

$$\mathbb{F} = \{\psi_{\boldsymbol{\lambda}}(\mathbf{u}) \mid \boldsymbol{\lambda} \in \mathbb{L} \setminus \{\mathbf{0}\}\}, \tag{5}$$

where $\mathbf{u}$ is the m-dimensional spatial and/or time variable in which the each 1D filterbank is defined. By definition,

$$\mathbb{L} = (\Lambda_1^+ \cup \{0\}) \times (\Lambda_2^+ \cup \Lambda_2^- \cup \{0\}) \times ... \times (\Lambda_m^+ \cup \Lambda_m^- \cup \{0\}); \tag{6}$$

$$\Lambda_i^- = \left\{-\lambda \;\middle|\; \lambda \in \Lambda_i^+\right\}, \tag{7}$$

where $\cup$ indicates the set union operator and $\times$ the Cartesian product. For $m = 1$, the provided definitions result in a conventional 1D scattering transform [2].

The m-dimensional LPF is defined as

$$\phi(\mathbf{u}) = \psi_{\mathbf{0}}(\mathbf{u}). \tag{8}$$

III. Separable Scattering Transform

A. Transform

The scattering transform requires 2 steps to provide scattering coefficients. The scalogram operator $\mathcal{U}_j$ iteratively filters a discrete signal $x[\mathbf{n}]$ for a given set of filters $\mathbb{F}$, which is then averaged by the LPF. $\mathbf{n}$ represents a multidimensional index variable.

$$\mathcal{U}_j x[\mathbf{n}, \boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_j] = \left|(\mathcal{U}_{j-1} \ ... \ \mathcal{U}_1 x) * \psi_{\boldsymbol{\lambda}_j}\right|, \ \forall \ \psi_{\boldsymbol{\lambda}_j} \in \mathbb{F}; \tag{9}$$

$$\mathcal{U}_1 x[\mathbf{n}, \boldsymbol{\lambda}_1] = \left|x * \psi_{\boldsymbol{\lambda}_1}\right|, \ \psi_{\boldsymbol{\lambda}_1} \in \mathbb{F}. \tag{10}$$

The scattering operator $\mathcal{S}_j$ provides the output coefficients at the j'th order of the scattering transform:

$$\mathcal{S}_j x[\mathbf{n}, \boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_j] = \mathcal{U}_j x * \phi. \tag{11}$$

The LPF ϕ remains constant throughout the transform. Note that the modulus/magnitude operator $|\cdot|$ demodulates the output of the filters, effectively extracting the Hilbert envelope from a band-limited signal [13].

The j'th scattering order adds an additional axis of paths indexed by $\boldsymbol{\lambda}_j$. However, not all paths need to be evaluated, since some paths have smaller bandwidths, thereby requiring fewer filters to extract the information lost by averaging. In particular, we only evaluate paths in which the centre frequencies of all elements of the vector $\boldsymbol{\lambda}_j$ are smaller than their corresponding bandwidth of the previous path's filter $\psi_{\boldsymbol{\lambda}_{j-1}}$. Path pruning is therefore dependent on α and β.

B. Downsampling Strategy

Since each filter specified by $\boldsymbol{\lambda}$ has its own bandwidth, we can employ downsampling across all paths non-uniformly. However, some care is required to ensure the compounded downsampling steps across all paths result in a uniform sampling frequency of the output scattering coefficients.

Many applications are insensitive to small changes in d. As such, we propose a strategy to find an optimal d given a target and tolerance value. For some applications, choosing d such that the downsampling factor is a power of 2 is the simplest solution to achieve optimality.

Without prior knowledge of the filterbank configurations, given a target invariance scale of $\bar{d}$ samples and a tolerance ϵ, we can optimise $d \in \{\lfloor(1-\epsilon)\bar{d}\rfloor, \lceil(1+\epsilon)\bar{d}\rceil\}$ such that it results in largest number of supported downsampling configurations.

A downsample factor d which decomposes into a set of n prime factors $\{p_1, ..., p_n\}$ with a corresponding multiset $\mathbb{M} = \{m_1, ..., m_n\}$, where m_i is the multiplicity of the prime p_i. We can find an optimal d by maximising the sum $\sum_{m \in M} m$.

Morlet filters in a 1D filterbank may be downsampled by a factor d_{ψ_1} prior to low-pass filtering, and then downsampling again by a factor d_{ϕ_1} after low-pass filtering. As such, the compounded effect of downsampling restricts $d = d_{\psi_1} \cdot d_{\phi_1}$.

In the second order of scattering, the process is repeated with an additional pre-low-pass downsampling factor of d_{ψ_2}. The second level application of ϕ then downsamples by a factor d_{ϕ_2}. To maintain a consistent output sampling frequency, it restricts $d = d_{\psi_2} \cdot d_{\psi_1} \cdot d_{\phi_2}$.

Continuing the downsampling chain, the i'th level of downsampling requires $d = d_{\psi_i} \cdot ... \cdot d_{\psi_1} \cdot d_{\phi_i}$. The output of each operation of $\mathcal{U}_i$ and $\mathcal{S}_i$ must be downsampled as much as possible in order to make subsequent operations faster. To ensure that the application of all the downsampling steps are efficient, we require d to have as many prime factors as possible, including factor multiplicity, so that a larger variety of downsampling combinations may be supported.

Consider a 1D wavelet filterbank and a single wavelet filter $\psi_1[n]$, applied to a discrete-time signal x. The operations required to compute the scattering coefficients is then notated for simplicity as

$$u_1[n] = \left|x * \psi_1\right|; \tag{12}$$

$$s_1[n] = y_1 * \phi. \tag{13}$$

The bandwidth of u_1 is the bandwidth of an arbitrary first filter ψ_1. This follows from the Hilbert envelope computed by the analytic wavelet filter ψ_λ and the modulus $|\cdot|$. The bandwidth of s_1 is the bandwidth of ϕ.

Critical downsampling of a wavelet ψ with a bandwidth of σ_ω is achieved by a factor of $d_\psi = \left\lfloor \frac{\pi}{\beta\sigma_\omega} \right\rfloor$. Critical downsampling of s_1, is achieved via a factor of d, by definition.

We can efficiently compute s_1 using compounded downsampling steps:

$$(s_1)_{\downarrow d} = \left((u_1)_{\downarrow d_1} * (\phi)_{\downarrow d_1}\right)_{\downarrow d_2}, \tag{14}$$

such that $d = d_1 \cdot d_2, \ d_1, d_2 \in \mathbb{N}^+$, with $d_1 | d$ and $d_1 \leq d_{\psi_1}$. In order to find d_1, we decrement d_{ψ_1} until it divides d evenly. Each scalogram $u_j[n]$ is not necessarily downsampled optimally, but has a downsampling factor which guarentees a consistent scattering coefficient output sample frequency.

A second order of scattering with a filter path of (ψ_1, ψ_2) is performed on the downsampled u_1:

$$u_2 = \left|(u_1)_{\downarrow d_1} * (\psi_2)_{\downarrow d_1}\right|; \tag{15}$$

$$(s_2)_{\downarrow d} = \left((u_2)_{\downarrow d_3} * (\phi)_{\downarrow d_3}\right)_{\downarrow d_4}, \tag{16}$$

such that $d = d_1 \cdot d_3 \cdot d_4, \ d_1, d_3, d_4 \in \mathbb{N}^+, d_3 \leq d_{\psi_2}, d_3 | \frac{d}{d_1}$. The proposed downsampling scheme can be extended to an arbitrary number of scattering orders.

C. Convolutions

Optimal fast Fourier transform (FFT) convolutions can be achieved by performing downsampling in the frequency domain instead of the time domain. It is straightforward to verify that $|x * \psi|_{\downarrow r} = |(x * \psi)_{\downarrow r}|$, since the modulus is an element-wise operation, for some downsampling factor r. Given the signal and filter Fourier transforms $\hat{x}$ and $\hat{\psi}$, we then have

$$x \otimes \psi[n] \xleftrightarrow{\mathcal{FFT}} \hat{x} \cdot \hat{\psi}[k], \tag{17}$$

where the $\otimes$ operator represents a circular convolution and k is the frequency index.

Given that $r | N$, we can express (17) when downsampled as a periodised summation [13] in the frequency domain

$$(x \otimes \psi[n])_{\downarrow r} \xleftrightarrow{\mathcal{FFT}} \frac{1}{r} \sum_{i=0}^{r-1} \hat{x} \cdot \hat{\psi}[k + iN/r], \tag{18}$$

TABLE I
MEDMNIST CLASSIFICATION RESULTS OF COMPARED TO BASELINE NN APPROACHES ([12])

Methods	Organ ($c=11$)		Nodule ($c=2$)		Fracture ($c=3$)		Adrenal ($c=2$)		Vessel ($c=2$)		Synapse ($c=2$)	
	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
ResNet-1810+2.5D	0.977	0.788	0.838	0.835	0.587	0.451	0.718	0.772	0.748	0.846	0.634	0.696
ResNet-1810+3D	0.996	0.907	0.863	0.844	0.712	0.508	0.827	0.721	0.874	0.877	0.82	0.745
ResNet-1810+ACS41	0.994	0.900	0.873	0.847	0.714	0.497	0.839	0.754	0.930	0.928	0.705	0.722
ResNet-5010+2.5D	0.974	0.769	0.835	0.848	0.552	0.397	0.732	0.763	0.751	0.877	0.669	0.735
ResNet-5010+3D	0.994	0.883	0.875	0.847	0.725	0.494	0.828	0.745	0.907	0.918	**0.851**	**0.795**
ResNet-5010+ACS41	0.994	0.889	0.886	0.841	**0.750**	**0.517**	0.828	0.758	0.912	0.858	0.719	0.709
auto-sklearn11	0.977	0.814	**0.914**	**0.874**	0.628	0.453	0.828	**0.802**	0.910	**0.915**	0.631	0.730
AutoKeras12	0.979	0.804	0.844	0.834	0.642	0.458	0.804	0.705	0.773	0.894	0.538	0.724
Separable WS + NN (Ours)	**0.998**	**0.941**	0.858	0.797	0.614	0.458	**0.875**	0.792	**0.962**	0.895	0.715	0.525

TABLE II
MNIST CLASSIFICATION ERROR RATE (%) OF SEPARABLE AND 2D SCATTERING COEFFICIENTS USING A NN CLASSIFIER

2D WS + NN	$l=1,\ J=2$	0.64 ± 0.05
	$l=2,\ J=3$	0.50 ± 0.03
Separable WS + NN	$l=1,\ d=(4,4)$	0.63 ± 0.05
	$l=2,\ d=(4,4)$	0.52 ± 0.04

which can be efficiently implemented via shape manipulation of tensors in computational packages like MATLAB or PyTorch [14]. To compute valid convolutions, we must pad x and ψ to have a total length of $N = N_x + d + c$, where $c \in \mathbb{N}^+$ is a constant that ensures that $d|N$.

IV. RESULTS

A. MNIST

The MNIST dataset [11] contains 60000 training and 10000 test samples. Unlike in [1], which decorrelates scattering coefficients with a discrete cosine transform prior to classification, we perform classification on the scattering coefficients directly. Kymatio [9] is used to produce the 2D scattering coefficients. Our implementation of separable scattering is implemented similarly to Kymatio, with PyTorch [14] as a backend for FFT convolutions and NN models.

Unless specified otherwise, all experiments have $\beta = \alpha = 2.5$. Scattering features are normalised prior to classification, according to the mean and SD calculated on the training set. No data augmentation is performed.

Due to its separability and non-angularly spaced filters, separable scattering does not perform as well compared to 2D scattering when using simple classifiers, such as linear discriminant analysis (LDA) [15]. To illustrate that this performance discrepancy is not of significant consequence in a NN setting, we test performance on the full dataset utilising a simple architecture. The neural network architecture used has an input layer with 256 neurons, followed by two hidden layers with 128 and 64 neurons respectively. The output layer has 10 neurons - one for each digit. Input and hidden layers are followed by a batch norm layer [16] and ReLU activation function [17]. The output layer is followed by a softmax function. The Adam optimiser [18] is used with cross-entropy loss, a batch size of 256 and learning rate of $3 \cdot 10^{-5}$. 5000 of the 60000 training samples are reserved for validation and removed from the training set. Training is stopped when validation loss starts to increase. The NN model is initialised with random weights, and the experiment is repeated 50 times. Different invariance scales were tested, and the best results are reported in table II. Tests are repeated for $l \in \{1, 2\}$ levels of scattering.

B. MedMNIST3D

The MedMNIST3D datasets are a subset of the MedMNIST dataset group [12], where each 3D dataset contains $28 \times 28 \times 28$ images with 2, 3 or 11 classes. Train, test and validation data partitions are provided by the authors. All datasets have on the order of 1000 training samples. We compare the baseline NN results provided in [12] with separable scattering features combined with a simple NN classifier.

We use NN classifier with an input layer containing 1024 neurons, followed by two hidden layers with 512 and 256 neurons respectively. Input and hidden layers layers are each followed by a batch norm layer [16] and a ReLU non-linearity [17]. For datasets with two classes, the output layer is a single neuron followed by a sigmoid activation function. For datasets with more than two classes, the output layer has a size equal to the number of classes, followed by a softmax activation. Binary cross-entropy loss are used for datasets with two classes, otherwise cross-entropy loss is used. The Adam optimiser [18] with a learning rate of $1 \cdot 10^{-5}$ is used. All other configuration parameters are identical to the model used for the MNIST dataset.

A single level of scattering coefficients are computed, with $Q = (2, 2, 2)$ and $d = (4, 4, 4)$. Many MedMNIST3D datasets tend to be unbalanced, implying that area under curve (AUC) is a more reliable metric to measure model performance. The results are shown in table I, in which SOTA AUC performance is achieved for the Organ, Adrenal and Vessel datasets. Table I indicates the number of classes (c) for each of the datasets, with accuracy (ACC) also shown for reference. The performance of our method on non-SOTA results are comparable to the other baseline NN approaches presented in [12]. It is likely that better results can be achieved by the proposed method if the filters are made learnable and/or scattering parameters are uniquely optimised for each dataset.

REFERENCES

[1] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1872–1886, 2013.

[2] J. Andén and S. Mallat, "Multiscale scattering for audio classification.," in *ISMIR*, pp. 657–662, Miami, Florida, 2011.

[3] Y. Y. Tang, Y. Lu, and H. Yuan, "Hyperspectral image classification based on three-dimensional scattering wavelet transform," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, pp. 2467–2480, 5 2015.

[4] M. Eickenberg, G. Exarchakis, M. Hirn, and S. Mallat, "Solid harmonic wavelet scattering: Predicting quantum molecular energy from invariant descriptors of 3d electronic densities," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[5] S. Mallat, "Group invariant scattering," *Communications on Pure and Applied Mathematics*, vol. 65, no. 10, pp. 1331–1398, 2012.

[6] J. Andén, V. Lostanlen, and S. Mallat, "Joint time-frequency scattering," 7 2018.

[7] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *2018 IEEE spoken language technology workshop (SLT)*, pp. 1021–1028, IEEE, 2018.

[8] Y. Mao, Z. He, Z. Ma, X. Tang, and Z. Wang, "Efficient convolution neural networks for object tracking using separable convolution and filter pruning," *IEEE Access*, vol. 7, pp. 106466–106474, 2019.

[9] M. Andreux, T. Angles, G. Exarchakis, R. Leonarduzzi, G. Rochette, L. Thiry, J. Zarka, S. Mallat, J. Andén, E. Belilovsky, J. Bruna, V. Lostanlen, M. Chaudhary, M. J. Hirn, E. Oyallon, S. Zhang, C. Cella, and M. Eickenberg, "Kymatio: Scattering transforms in python," *Journal of Machine Learning Research*, vol. 21, no. 60, pp. 1–6, 2020.

[10] MathWorks Inc., "MATLAB Wavelet Toolbox (R2024a)," 2022.

[11] Y. LeCun, C. Cortes, C. Burges, *et al.*, "Mnist handwritten digit database," 2010.

[12] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification," *Scientific Data*, vol. 10, no. 1, p. 41, 2023.

[13] J. Kovacevic and M. Vetterli, *Wavelets and Subband Coding*. Prentice-Hall signal processing series, Prentice Hall PTR, 1995.

[14] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.

[15] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015.

[17] A. F. Agarap, "Deep learning using rectified linear units (relu)," 2019.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.