DMEL: THE DIFFERENTIABLE LOG-MEL SPECTROGRAM AS A TRAINABLE LAYER IN NEURAL NETWORKS

John Martinsson[⋆†] *Maria Sandsten*[⋆]

[⋆] Lund University, Centre for Mathematical Sciences, Lund, Sweden
[†] RISE Research Institutes of Sweden, Computer Science, Gothenburg, Sweden

ABSTRACT

In this paper we present the differentiable log-Mel spectrogram (DMEL) for audio classification. DMEL uses a Gaussian window, with a window length that can be jointly optimized with the neural network. DMEL is used as the input layer in different neural networks and evaluated on standard audio datasets. We show that DMEL achieves a higher average test accuracy for sub-optimal initial choices of the window length when compared to a baseline with a fixed window length. In addition, we analyse the computational cost of DMEL and compare to a standard hyperparameter search over different window lengths, showing favorable results for DMEL. Finally, an empirical evaluation on a carefully designed dataset is performed to investigate if the differentiable spectrogram actually learns the optimal window length. The design of the dataset relies on the theory of spectrogram resolution. We also empirically evaluate the convergence rate to the optimal window length.

***Index Terms*—** Deep learning, STFT, learnable Mel spectrogram, audio classification, adaptive transforms

1. INTRODUCTION

An increasing interest for using time-frequency images for feature extraction is seen in classification of audio data, typically human speech, music, and bioacoustics recordings. In audio classification, the spectrogram, the squared magnitude of the short-time Fourier transform (STFT), is typically mapped onto the Mel-scale using a Mel-filterbank [1]. This is called the Mel-spectrogram, which is then used as input to a neural network model.

The choice of Mel-filterbank affects the frequency resolution and additionally the choice of window length of the STFT creates a trade-off between time- and frequency resolution. Different trade-offs may be optimal for different audio classification tasks. Recent work has proposed the differentiable STFT (DSTFT) [2, 3, 4] where the window length, and thereby the time-frequency (TF) resolution, can be jointly optimized with the neural network. In [2] the DSTFT is proposed using a 50% fixed-overlap STFT and a Gaussian window. In [3] the constraint on the window function being Gaussian is relaxed, and a theory for a family of differentiable STFTs is presented. In these methods, the number of frequency bins is proportional to the window length, which can lead to high computational demands. Evaluations on complex audio classification tasks is therefore limited.

Recent work on learnable Mel-spectrograms include learning the Mel-filterbank [5], the energy normalization [6], and combinations of both [7, 8]. In this way, the feature extraction method can be optimized for the audio classification task at hand. Applications are seen in speech processing [9, 10], bird acoustic classification [11], and underwater acoustic classification [12].

In this work, we propose DMEL, the differentiable log-Mel spectrogram, which is an extension of DSTFT. DMEL is evaluated in a state-of-the-art convolutional neural network (CNN) for audio classification on complex audio datasets. We analyse the computational cost of DMEL and we also investigate the classification accuracy as well as the convergence rate for a simplified case. This is a step towards closing the gap between the DSTFT and the recent work using trainable filter-banks and normalization in the Mel-spectrogram for audio classification.

2. DMEL: DIFFERENTIABLE LOG-MEL SPECTROGRAM

The spectrogram is defined as

$$S_{x,\lambda}(t,f) = |F(t,f)|^2 = \\ = |\int_{-\infty}^{\infty} x(s-t)h(s)\exp(-i2\pi fs)ds|^2 \quad (1)$$

where $F(t,f)$ is the short-time Fourier transform (STFT) of the signal $x(t)$ using a Gaussian window

$$h(t) = \exp(-\frac{t^2}{2\lambda^2}), \quad (2)$$

with scaling parameter λ which controls the window length and thereby the TF resolution of the spectrogram.

Thanks to the Swedish Foundation for Strategic Research for funding.
Code: https://github.com/johnmartinsson/differentiable-mel-spectrogram

The STFT is differentiable with respect to the window parameter λ according to

$$\frac{dF(t,f)}{d\lambda} = \int_{-\infty}^{\infty} x(s-t)\frac{dh(s)}{d\lambda}\exp(-i2\pi fs)ds, \quad (3)$$

and a loss function $\mathcal{L}$ is differentiable w.r.t λ through gradient backpropagation using

$$\frac{d\mathcal{L}}{d\lambda} = \sum_{n=1}^{N}\sum_{k=0}^{K-1}\frac{d\mathcal{L}}{dF(n,k)}\frac{dF(n,k)}{d\lambda}, \quad (4)$$

where $F(t,f)$ is discretized to $F(n,k)$ with a fixed number of N bins in time and K bins in frequency [3]. Equations (1)-(4) define the differentiable spectrogram (DSPEC).

The model layer studied in this paper is the differentiable log-Mel spectrogram (DMEL), which is a novel extension where a set of Mel-filters $\{\psi_m\}_{m=1}^{M}$ are applied to DSPEC to map it to the Mel-scale

$$M_{x,\lambda}(n,m) = \log(\sum_{k=0}^{K-1} S_{x,\lambda}(n,k)\psi_m(k) + \epsilon). \quad (5)$$

The Mel-filters are defined as in [1] and $\epsilon = 1e^{-10}$ to avoid the logarithm of zero. The log-Mel filterbank preserves the gradients during backpropagation giving a log-Mel spectrogram with a trainable window size. We also let

$$l_\lambda = 1000 * 6 * \lambda / F_s, \quad (6)$$

denote the window length in milliseconds (ms), where F_s is the sample rate, and the factor 1000 converts to ms.

3. AUDIO DATA EXPERIMENTS

In this section we present the models and data used to evaluate DMEL, and the experiment setup. The results are then presented for each dataset and compared to the baseline. The baseline is the same model, but using a log-Mel spectrogram with a fixed window length as input layer instead of DMEL.

3.1. Audio data and models

DMEL is evaluated together with a linear model called "LNet" consisting of a linear layer followed by a softmax normalization, and a state-of-the-art convolutional neural network called "CNN6" from the PANNs family [13] detailed in table 1. We set the number of Mel-filters to $M = 64$ and the fixed hop length in the STFT is set to 10 ms. This imposes a bound on the achievable TF resolution, but is necessary to make the optimization feasible.

We evaluate DMEL in these two models on the audio MNIST dataset (A-MNIST) [14] and the ESC50 dataset [15]. The A-MNIST dataset consists in total of $30,000$ recordings collected from 60 different people speaking the numbers 0 to 9. The ESC50 dataset consists in total of $2,000$ recordings collected from 50 different environmental sound classes. All audio recordings are downsampled to $8,000$ Hz.

Table 1. The PANNs 6 layer convolutional neural network (CNN6) architecture.

Model	CNN6
Input	DMEL / baseline, 64 Mel bins
Conv. layers	(3x3 @ 64, BN, ReLU) x 2
	(3x3 @ 128, BN, ReLU) x 2
	(3x3 @ 256, BN, ReLU) x 2
	(3x3 @ 512, BN, ReLU) x 2
	Global average pooling
	FC 512, ReLU
Output	FC 50, Sigmoid

3.2. Experiments and results

The models are trained using the Adam optimizer for 100 epochs and the model with the lowest validation loss is chosen. The loss function is the standard cross-entropy loss. The task is to predict the ground truth class given the audio recording. For A-MNIST the recordings are split $60\%/20\%/20\%$ into training/validation/test datasets and for the ESC50 dataset the split is $70\%/10\%/20\%$. All parameter learning rates are 0.0001, except for λ, which is 1. The "CNN6" model was designed using a window length of 35 ms for general audio tasks [13], we therefore evaluate three different initial window lengths $l_{\lambda_{init}} \in \{10, 35, 300\}$ ms, to see if DMEL makes the model robust against this parameter choice. We do this 10 times for each model and hyper parameter configuration.

In table 2 we present the average test accuracy for the "CNN6" model on the ESC50 dataset when either using DMEL or the baseline as input layer to the model. The learned window length, denoted $l_{\lambda_{est}}$, is presented as min and max in the table. Note that the comparison is done pairwise between DMEL and the baseline for different initial window lengths, and that we do not expect DMEL to outperform the baseline when $l_{\lambda_{init}}$ is already suitable for the classification task. We use bold-face to indicate a significant difference. DMEL outperforms the baseline for the (presumably) sub-optimal choices $l_{\lambda_{init}} = 10$ ms and $l_{\lambda_{init}} = 300$ ms, and achieves similar results for $l_{\lambda_{init}} = 35$ ms (a typical choice

Model	$l_{\lambda_{init}}$	$l_{\lambda_{est}}$ (min, max)	Method	Accuracy
CNN6	10 ms	(25, 27) ms	**DMEL**	**87.3** $\pm$ 1.0
CNN6	10 ms	—	baseline	84.2 ± 1.2
CNN6	35 ms	(31, 90) ms	DMEL	86.1 ± 1.3
CNN6	35 ms	—	baseline	86.9 ± 0.7
CNN6	300 ms	(117, 153) ms	**DMEL**	**85.8** $\pm$ 1.2
CNN6	300 ms	—	baseline	84.7 ± 1.1

Table 2. Pairwise comparison between DMEL and the baseline for different $l_{\lambda_{init}}$ on the ESC50 dataset.

Model	$l_{\lambda_{init}}$	$l_{\lambda_{est}}$ (min, max)	Method	Accuracy
LNet	10 ms	(314, 442) ms	**DMEL**	**94.9** $\pm$ 1.0
LNet	10 ms	—	baseline	89.3 $\pm$ 1.0
LNet	35 ms	(398, 484) ms	**DMEL**	**95.0** $\pm$ 0.8
LNet	35 ms	—	baseline	91.9 $\pm$ 1.2
LNet	300 ms	(516, 608) ms	DMEL	95.3 $\pm$ 0.6
LNet	300 ms	—	baseline	95.3 $\pm$ 0.8

Table 3. Pairwise comparison between DMEL and the baseline for different $l_{\lambda_{init}}$ on the A-MNIST dataset.

for audio data). As a reference, the accuracy of the PANNs "CNN14" model, using a 35 ms window, is 83.3% when trained from scratch on ESC50 in the original paper [13].

In table 3 we present the average test accuracy for the "LNet" model on the A-MNIST dataset for different initial window lengths. A high accuracy is achieved for surprisingly large window lengths. DMEL learns this, achieving a high accuracy for all initial window lengths, significantly outperforming the baseline for $l_{\lambda_{init}} = 10$ ms and $l_{\lambda_{init}} = 35$ ms.

The results show that DMEL makes the audio classification model more robust to the choice of the initial window length, by adapting the window length to the task at hand. We note that DMEL introduces redundancy in the TF image due to the constant hop length, and in the following section we analyse the cost of DMEL.

4. COMPUTATIONAL COST ANALYSIS OF DMEL

The complexity of a fast Fourier transform (FFT) is $\mathcal{O}(L \log L)$, where $L = 6\lambda$ is the window size in samples. In the STFT, the FFT is applied N/c times, where N is signal length and c is hop size, which results in a TF image of pixel-size $n = MN/c$. The computational complexity of a CNN is $\mathcal{O}(n)$.

As baseline, we choose $c = L/2$, avoiding redundant information in the TF image, and we assume that a hyperparameter search is done linearly between 20 ms and 300 ms over D different window sizes, thus $n = 2MN/L$ where $M = 64$ is the number of Mel-bands. Using the cost constant C_1 for the FFT and the cost constant C_2 for the CNN we can derive the following computational cost expression for the baseline

$$C_{\text{baseline}} = BC_1 \sum_{i=1}^{D} N \log L_i + BC_2 \sum_{i=1}^{D} \frac{2MN}{L_i}, \quad (7)$$

where L_i is the different window lengths of the hyperparameter search and $B = |l_{\lambda_{opt}} - l_{\lambda_{init}}|/\alpha$, with $\alpha = 0.001$, is the assumed steps needed until convergence to the optimal window length $l_{\lambda_{opt}} = 35$ ms. For DMEL we do not need to train D different models, but need to set the hop size $c = 80$ (10 ms) to a constant, resulting in the cost expression

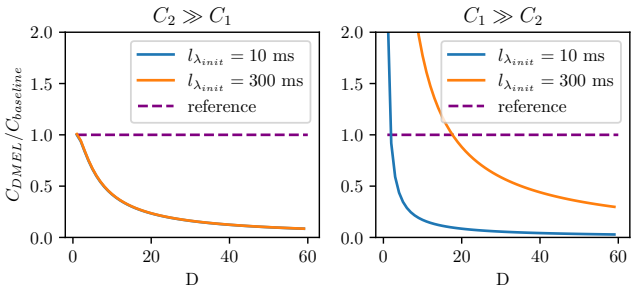

Fig. 1. The computational cost quotient $C_{\text{DMEL}}/C_{\text{baseline}}$ with respect to D, for two different $l_{\lambda_{init}}$, when $C_2 \gg C_1$ (left; blue and orange lines are overlapping) and $C_1 \gg C_2$ (right).

$$C_{\text{DMEL}} = C_1 \sum_{i=1}^{B} \frac{NL_i}{c} \log L_i + \frac{BC_2MN}{c}. \quad (8)$$

The relation $C_{\text{DMEL}}/C_{\text{baseline}}$ is independent of N and is depicted for different D in figure 1. When the computational cost is dominated by the neural network, $C_2 \gg C_1$, we see computational benefits for growing D, e.g. we see that DMEL requires half the computational cost compared to baseline for $D \approx 10$. In the case when $C_1 \gg C_2$, when the cost is dominated by the FFT, we see the highest reduction for a short initial window in DMEL (blue line).

5. EVALUATION OF CLASSIFICATION ACCURACY AND CONVERGENCE RATE

To investigate classification accuracy, we present a synthetic dataset, for which the accuracy should be directly dependent on the TF resolution, i.e. the window length. This is a simplified analysis and we therefore use DSPEC instead of DMEL to search for the scaling parameter. We will also verify the findings from a theoretical aspect and investigate how the convergence rate depends on the initial window length.

5.1. Simulated dataset

The simulated dataset consists of three classes, class 1, a single Gaussian-pulse, class 2 and 3, with two pulses separated either in time or frequency, as exemplified in figure 2. A Gaussian-pulse is defined as

$$g(n_0, f_0, \sigma) = A \exp(-\frac{(n-n_0)^2}{2\sigma^2}) \sin(2\pi f_0 n + \phi), \quad (9)$$

where the parameters are chosen to give TF symmetry ($\sigma = 6.4$) and optimal TF separation for an optimal window length ($\lambda = 6.4$), see section 5.3. Gaussian noise is added to all signal classes and A, ϕ, n_0 and f_0 are chosen at random. Full parameter description is given in source code (page 1). In

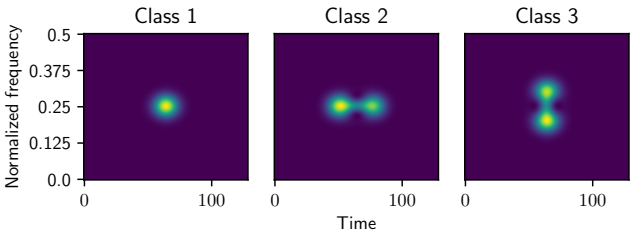

Fig. 2. Examples from the simulated Gaussian-pulse dataset.

total $5,000$ samples is used, with approximately $1/3$ of each class. Using $N = 128$ with hop size one and $K = 256$, the resulting TF images are of square size $N \times K/2$.

5.2. Experiment and results

We evaluate "LNet" on this dataset using either DSPEC or a fixed window length baseline, which is trained in the same way as the audio classification model, with the exception that Stochastic Gradient Descent (SGD) is used.

The λ_{init} which on average give the highest test accuracy for both methods is $\lambda_{init} = \sigma = 6.4$ (see table 4). DSPEC is able to learn a λ_{est} close to 6.4 for all λ_{init}, significantly outperforming the baseline for the sub-optimal choices.

5.3. Theory on time-frequency resolution and symmetry

In this subsection we derive the parameter choices for optimal TF resolution and TF symmetry. The reasoning relies on that the resolution limit of two closely spaced Gaussian functions is two times the actual Gaussian function scaling parameter, [16]. The window is Gaussian with parameter λ and the signal is $g(0, 0, \sigma)$, which can be generalized to any n_0, f_0 due to TF shift invariance [17]. The resulting discretized spectrogram is

$$S_x(n,k) = \frac{2\lambda^2\sigma^2\pi}{\lambda^2+\sigma^2}\exp(-\frac{1}{2}(\frac{n}{\delta_t})^2 - \frac{1}{2}(\frac{k}{\delta_f})^2), \qquad (10)$$

a two-dimensional Gaussian function with scaling parameters

$$\delta_t = \sqrt{\frac{\lambda^2+\sigma^2}{2}}, \quad \delta_f = \frac{K}{2\pi\lambda\sigma}\sqrt{\frac{\lambda^2+\sigma^2}{2}}. \qquad (11)$$

Model	λ_{init}	λ_{est} (min, max)	Method	Accuracy
LNet	1.3	(4.8, 5.6)	**DSPEC**	**98.5** $\pm$ 0.2
LNet	1.3	—	baseline	95.5 $\pm$ 0.2
LNet	6.4	(5.3, 6.0)	DSPEC	98.5 $\pm$ 0.2
LNet	6.4	—	baseline	98.8 $\pm$ 0.3
LNet	31.9	(3.4, 6.5)	**DSPEC**	**98.0** $\pm$ 0.4
LNet	31.9	—	baseline	94.9 $\pm$ 0.4

Table 4. Pairwise comparison between DSPEC and the baseline for different $l_{\lambda_{init}}$ on the Gaussian-pulse dataset.

Table 5. Convergence rate experiment

λ_{init}	1.3	31.9
Iterations	75.5 ± 1.3	186.0 ± 37.8

Minimizing δ_t and δ_f will result in optimal TF resolution. The corresponding TF cross-section area is $A = \pi\delta_t\delta_f$ with derivative as

$$\frac{dA}{d\lambda} = \frac{K\sigma}{4}(\frac{1}{\sigma^2} - \frac{1}{\lambda^2}), \qquad (12)$$

giving $\lambda_{opt} = \sigma$ and $A_{min} = K/2 = 128$. Optimal TF resolution is therefore given using a matched window [17]. For TF symmetry (equal number of bins in time- and frequency) we also set $\delta_t = \delta_f$ in (11), and we find for the matched window case, $\lambda_{opt} = \sigma = \sqrt{K/(2\pi)} = 6.4$.

5.4. Convergence rate

Relying on TF symmetry we perform an experiment on the difference in convergence rate for λ when approaching the optimal solution from a small initial value, or a large initial value of λ. We compute the optimally concentrated spectrogram for a Gaussian-pulse signal and use this as the ground truth (see leftmost image in figure 2). The task is to learn this spectrogram (i.e., the true value λ_{opt}) using DSPEC with SGD and a mean-squared error (MSE) loss. The MSE loss is between the estimated spectrogram and the ground truth. We study two carefully chosen values for λ_{init}. Both give exactly the same cross-section area A and therefore also the same initial MSE loss. We then measure the number of SGD iterations until $|\lambda_{est} - \lambda_{opt}| < 0.1$, and present the average number of iterations until convergence over 50 signals in table 5. The convergence rate is twice as fast when λ_{init} is chosen as the value smaller than λ_{opt}. Faster convergence means a smaller B in (8), leading to further reduction in computational cost.

6. CONCLUSIONS

We introduce DMEL: a differentiable log-Mel spectrogram for audio classification, allowing joint optimization of the window length and the neural network. DMEL achieves a higher test accuracy on average than the baseline with a fixed window length for all non-optimal initial window lengths on all evaluated datasets. In addition, we show that DMEL leads to a reduced computational cost compared to a standard hyperparameter search over different window lengths, especially if the initial window length is short. An empirical evaluation shows that the differentiable spectrogram is able to learn the optimal window length on a carefully designed classification task. Finally, a convergence rate experiment indicates that a shorter window is beneficial for fast convergence. The overall results suggest that it is favorable to initialize the DMEL with a short window, resulting in lower computational cost and faster convergence.

7. REFERENCES

[1] Malcolm Slaney, "Auditory toolbox," Tech. Rep., Interval Research Corporation, 1998.

[2] An Zhao, Krishna Subramani, and Paris Smaragdis, "Optimizing short-time Fourier transform parameters via gradient descent," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2021-June, no. 2, pp. 736–740, 2021.

[3] Maxime Leiber, Axel Barrau, Yosra Marnissi, and Dany Abboud, "A differentiable short-time Fourier transform with respect to the window length," in *2022 30th European Signal Processing Conference (EUSIPCO)*, 2022, pp. 1392–1396.

[4] Maxime Leiber, Yosra Marnissi, Axel Barrau, and Mohammed El Badaoui, "Differentiable Adaptive Short-Time Fourier Transform with Respect to the Window Length," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[5] Neil Zeghidour, Nicolas Usunier, Iasonas Kokkinos, Thomas Schaiz, Gabriel Synnaeve, and Emmanuel Dupoux, "Learning Filterbanks from Raw Speech for Phone Recognition," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2018-April, pp. 5509–5513, 2018.

[6] Yuxuan Wang, Pascal Getreuer, Thad Hughes, Richard F. Lyon, and Rif A. Saurous, "Trainable frontend for robust and far-field keyword spotting," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, , no. 1, pp. 5670–5674, 2017.

[7] Neil Zeghidour, Olivier Teboul, Félix de Chaumont Quitry, and Marco Tagliasacchi, "LEAF: A Learnable Frontend for Audio Classification," in *ICLR, International Conference on Learning Representations*, 2021, pp. 1–16.

[8] Jan Schlüter and Gerald Gutenbrunner, "EfficientLEAF: A Faster LEarnable Audio Frontend of Questionable Use," *European Signal Processing Conference*, vol. 2022-Augus, pp. 205–208, 2022.

[9] Miguel Arjona Ramirez, Wesley Beccaro, Demostenes Zegarra Rodriguez, and Renata Lopes Rosa, "Differentiable Measures for Speech Spectral Modeling," *IEEE Access*, vol. 10, pp. 17609–17618, 2022.

[10] Quchen Fu, Zhongwei Teng, Jules White, Maria E. Powell, and Douglas C. Schmidt, "Fastaudio: a Learnable Audio Front-End for Spoof Speech Detection," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2022-May, pp. 3693–3697, 2022.

[11] Mark Anderson and Naomi Harte, "Learnable Acoustic Frontends in Bird Activity Detection," in *International Workshop on Acoustic Signal Enhancement, IWAENC 2022 - Proceedings*, 2022.

[12] Jiawei Ren, Yuan Xie, Xiaowei Zhang, and Ji Xu, "UALF: A learnable front-end for intelligent underwater acoustic classification system," *Ocean Engineering*, vol. 264, no. September, 2022.

[13] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 28, pp. 2880–2894, nov 2020.

[14] Sören Becker, Marcel Ackermann, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek, "Interpreting and explaining deep neural networks for classification of audio signals," *CoRR*, vol. abs/1807.03418, 2018.

[15] Karol J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM International Conference on Multimedia*, New York, NY, USA, 2015, MM '15, p. 1015–1018, Association for Computing Machinery.

[16] Hajo Holzmann and Sebastian Vollmer, "A likelihood ratio test for bimodality in two-component mixtures with application to regional income distribution in the EU.," *AStA Advances in Statistical Analysis: A Journal of the German Statistical Society*, vol. 92, no. 1, pp. 57 – 69, 2008.

[17] Leon Cohen, *Time-Frequency Analysis*, Prentice-Hall, 1995.