Spectral trancoder : using pretrained urban sound classifiers on undersampled spectral representations

Modan Tailleur, Mathieu Lagrange, Pierre Aumond, Vincent Tourre

SPECTRAL TRANSCODER : USING PRETRAINED URBAN SOUND CLASSIFIERS ON UNDERSAMPLED SPECTRAL REPRESENTATIONS

Modan Tailleur[1]*, Mathieu Lagrange*[1]*, Pierre Aumond*[2]*, Vincent Tourre*[3]

[1] Nantes Université, École Centrale Nantes,
CNRS, LS2N, UMR 6004, F-44000 Nantes, France, {modan.tailleur,mathieu.lagrange}@ls2n.fr
[2] Univ Gustave Eiffel, CEREMA, UMRAE, F-44344 Bouguenais, France, pierre.aumond@univ-eiffel.fr
[3] Nantes Université, École Centrale Nantes, CNRS, AAU, UMR 1563, F-44000 Nantes, France,
vincent.tourre@ec-nantes.fr

ABSTRACT

Slow or fast third-octave bands representations (with a frame resp. every 1-s and 125-ms) have been a de facto standard for urban acoustics, used for example in long-term monitoring applications. It has the advantages of requiring few storage capabilities and of preserving privacy. As most audio classification algorithms take Mel spectral representations with very fast time weighting (ex. 10-ms) as input, very few studies have tackled classification tasks using other kinds of spectral representations of audio such as slow or fast third-octave spectra.

In this paper, we present a convolutional neural network architecture for transcoding fast third-octave spectrograms into Mel spectrograms, so that it could be used as input for robust pre-trained models such as YAMNet or PANN models. Compared to training a model that would take fast third-octave spectrograms as input, this approach is more effective and requires less training effort. Even if a fast third-octave spectrogram is less precise both on time and frequency dimensions, experiments show that the proposed method still allows for classification accuracy of 62.4% on UrbanSound8k and 0.44 macro AUPRC on SONYC-UST.

Index Terms— Convolutional Neural Network (CNN), Generative algorithm, third-octave spectrogram, Mel spectrogram, Urban soundscape

1. INTRODUCTION

In recent years, various sound source classification models have gained recognition for their robustness. Among them, YAMNet [1] and PANNs [2] pre-trained models have emerged as powerful models capable of predicting the presence of more than 500 sound sources, thanks to their training on the extensive Audioset database [3]. These models are widely recognized as among the most effective sound source classification models available and use Mel spectral representations with a frame every 10-ms as input .

IEC 61672-1 [4] standardizes the measurement of fast (125-ms) and slow (1-s) third-octave spectral representations, which have been used in several noise monitoring applications [5, 6, 7, 8, 9, 10]. Fast third-octave spectrograms offer several advantages over Mel spectrograms for long-term monitoring applications. First, they make recordings unintelligible and thus preserve privacy, as demonstrated by Gontier et al. [11]. Moreover, they are more lightweight, with a bit rate approximately 138 times lower than that of 16bits, 32kHz, mono waveform recordings and about 30 times lower than that of Mel recordings (see table 1 for precise references).

Gontier et al. [12] addressed multi-label classification tasks in urban environments using a Convolutional Neural Network (CNN) directly trained on third-octave spectrograms. While their model showed good performance on the Cense Lorient dataset [8], it lacks robustness on other third-octave recorded datasets. This limitation arises partly from training the model on highly homogeneous datasets. Pre-trained models such as YAMNet and PANNs, on the other hand, have shown robustness in a variety of sound source classification tasks. Unfortunately, these models are trained on Mel spectrograms with 10-ms frames, and can only consider the corresponding Mel representation as input.

To enable the direct use of those pre-trained models with other types of spectrograms such as fast third-octave ones, we present in this paper a transcoding method that converts fast third-octave spectrograms into Mel spectrograms. This transcoding operation is done using a CNN module learned with a teacher-student approach that leverages the pre-trained models' outputs to reconstruct Mel spectrograms. While this study focuses on a specific fast third-octave representation, we believe that the proposed method can be adapted to any kind of spectral representation. Section 2 reviews prior work on the transcoding task. Sections 3 and 4 outline our model architecture and training method. In section 5, we evaluate the performances of the transcoder. Generated audio and open source code are available online. [1]

2. RELATED WORK

To the best of our knowledge, no work is available for the task at hand in audio processing specifically. In computer vision, several methods have been proposed to address the task of converting one set of features to another set of features (feature translation) [13, 14]. A pseudo-inverse can be employed to retrieve a Mel spectrogram from a fast third-octave spectrogram and temporal information can be interpolated. This would result in a blurred Mel spectrogram, which could be seen as analogous to a noisy image in a denoising paradigm. Auto-encoding methods [15], adversarial methods [16], and diffusion methods [17] have been used in super-resolution and denoising tasks.

In contrast to previous works, our goal is to obtain generated Mel spectrograms that can achieve similar output class distributions as the original Mel spectrogram when processed by the pre-trained model used for training the transcoder.

[1] Companion website: `https://github.com/modantailleur/paperSpectralTranscoder`

3. METHODS

3.1. Spectral representations

In this study, we selected the Lorient Cense project fast third-octave calculation method [8] which involves computing 29 third-octave bands within the frequency range of 20Hz to 12,5kHz, using a rectangular 125-ms temporal window.

It is worth noting that YAMNet and PANNs models require Mel spectrograms as input, but the spectrograms used by these classifiers differ slightly from each other (as shown in Table 1). Therefore, we present two different transcoders in the subsequent sections to match the input requirements of each pre-trained model.

spectral representation	Mel		Third-Octave
origin	PANN	YAMNet	Cense Lorient
sample rate	32kHz	16kHz	32kHz
window size	1024 (32ms)	400 (25ms)	4096 (128ms)
fft size	1024 (32ms)	512 (32ms)	4096 (128ms)
hop size	320 (10ms)	160 (10ms)	4000 (125ms)
window	hann	hann	rectangular
frequency bins	64	64	29
min frequency	50Hz	125Hz	20Hz
max frequency	14kHz	7,5kHz	12,5kHz
mel normalisation	slaney	-	-
mel formula	slaney	htk	-
log offset	1.0	0.001	19.95
bit rate	103kb/s	100kb/s	3,71kb/s

Table 1: Differences between PANN (ResNet38) and YAMNet Mel spectral spectrograms, and Cense third-octave spectrograms

3.2. Model

The proposed CNN transcoder model, consists of two parts: a PINV transcoder and a Convolutional Neural Network (CNN) (see Figure 2). The PINV transcoder presented in figure 1 first reconstructs the full-band spectrogram from the third-octave spectrogram using a pseudo-inverse method. Then, it performs time-axis interpolation to match the time dimension of the target Mel spectrogram. Finally, the log Mel filterbank is applied to the full-band spectrogram, resulting in a roughly predicted Mel spectrogram. This PINV transcoder conveniently matches the target Mel spectrogram dimensions, and is adaptable to various undersampled spectral data.

The CNN part then refines the Mel spectrogram by adding residual information to it (see figure 2). The CNN architecture, which is identical to the one used by Lagrange et al. [18], is fully convolutional and has several layers, each employing rectified linear units (ReLU) activations. In the following sections, we refer to our transcoder, which is trained on pre-trained models' output logits, as CNN trained on logits (or CNN-logits).

3.3. Teacher-student approach

We take a teacher-student approach to train our CNN model in order to generate a Mel spectrogram by taking into account the output of YAMNet or PANNs pre-trained classifiers (see figure 3). We selected the ResNet38 PANN model, which has 73,783,247 parameters, as it is the most performing model to date that uses Mel spectrograms as input [2]. We also consider the well-established YAMNet classifier which has a lower number of parameters: 3,740,425. PANNs and YamNet parameters are not updated during the CNN transcoder training, reducing computational complexity and ensuring broader applicability to pre-trained classifiers using similar Mel spectrogram inputs.

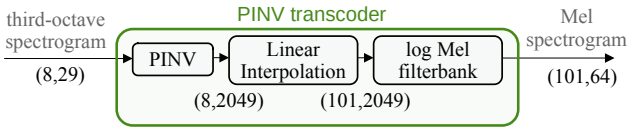

Figure 1: PINV transcoder architecture, to recover a 1s sample PANN Mel spectrogram from a 1s sample fast third-octave spectrogram

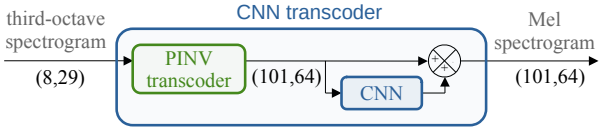

Figure 2: CNN transcoder architecture, to recover a 1s sample PANN Mel spectrogram from a 1s sample fast third-octave spectrogram

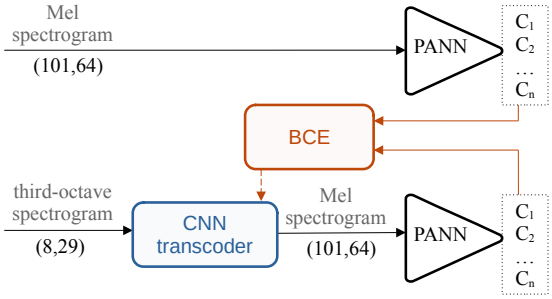

Figure 3: PANN CNN transcoder trained with a teacher-student approach (CNN-logits) using the Binary Cross-Entropy (BCE) loss function

4. EXPERIMENTAL PROTOCOL

4.1. Data

The dataset used for training and evaluating our models is the TAU Urban Acoustic Scenes 2020 Mobile dataset [19]. This dataset consists of 10-second audio clips from 10 different acoustic scenes, namely airport, indoor shopping mall, metro station, pedestrian street, public square, street with a medium level of traffic, traveling by tram, traveling by bus, traveling by an underground metro, and urban park. The dataset includes recordings from multiple devices that overlap in the given development subset. As the evaluation dataset has not been released yet, we use only the development subset for training and evaluating our models. To ensure non-overlapping data, we use only data from device A, which provides 29h20 of audio. We randomly split the development subset into training (75%), validation (12.5%), and evaluation (12.5%) sets. All audio files are normalized based on the maximum absolute value.

4.2. Baselines

In this study, we compare the performance of the CNN-logits transcoder with the performance of a reference PINV transcoder (as shown in Figure 1), which does not require any learning.

In addition, we explore an alternative training method that is solely based on the Mean Squared Error (MSE) loss between the generated Mel spectrogram and the ground truth spectrogram, without relying on a teacher-student approach. This transcoder will be referred to as CNN trained on mels (or CNN-mels) in the subsequent sections.

To further evaluate the performance of our proposed teacher-student approach, we compare it with other teacher-student methods that are not explicitly designed for transcoding fast third-octave spectrograms into Mel spectrograms (see figure 4). Specifically, we retrain the PANN and YAMNet models, as well as efficient nets (efficient net b_0 with 4,682,059 parameters and efficient net b_7 with 65,135,455 parameters) [20], using pseudo-inverted Mel-spectrograms as input with the method illustrated in Figure 1. In the subsequent sections, we will refer to these retrained models as PANN-1/3-oct, YAMNet-1/3-oct, Effnet-b0, and Effnet-b7.

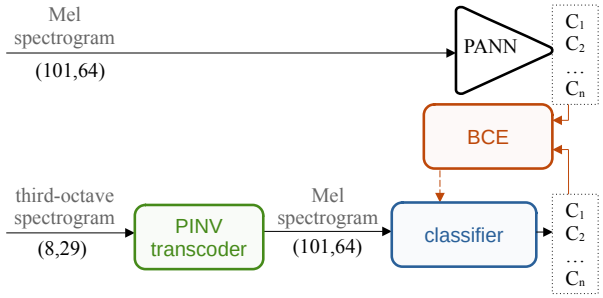

Figure 4: Classifier trained with a teacher-student approach, to match PANN outputs

4.3. Training procedure

Both types of training methods, i.e., with and without teacher-student approach, employ Adam optimizer [21] during optimization.

For the CNN architecture, we have conducted experiments with varying kernel sizes, numbers of layers, dilations, numbers of channels, and learning rates. Only the models with parameters leading to the best loss are presented in the subsequent sections.

All the models are trained for 200 epochs, with a batch size of 64, leading to 2,472,000 iterations. We checked empirically that convergence is reached for all models.

4.4. Metrics

To assess the performance of the proposed methods, we introduce the Prediction to Prediction accuracy on First Class (PtoPa-FC) metric, and calculate it on our evaluation subset of the TAU Urban Acoustic Scenes 2020 Mobile dataset. This metric measures the accuracy of the pre-trained model that uses transcoded Mel spectrograms as input in predicting the same first class as that of the pre-trained models that use ground truth Mel spectrograms as input. However, it should be noted that this metric only provides information regarding the accuracy of the first predicted class. Therefore, we also analyze the KL-divergence between the distribution of the two predictions vectors. All predictions are based on 10-second audio excerpts.

To further evaluate the effectiveness of our models, we subject them to testing on two additional annotated datasets: SONYC-UST [22] and UrbanSound8K [23]. However, the output classes of the pre-trained PANN and YAMNet models do not correspond exactly to the target classes of these datasets. To address this issue, we propose to augment the pre-trained models with two additional fully connected layers that have an intermediate size of 100. These layers are trained on the training subset of SONYC-UST and evaluated on the test subset, and we employ cross-validation for UrbanSound8K as recommended by the authors. The objective of the fully connected layers is to aggregate the 527 (or 521) input classes of the pre-trained models into the 8 (or 10) target classes of UrbanSound8K or SONYC-UST datasets, respectively. Importantly, we only train the additional fully connected layers, and the pre-trained models are not re-trained during this process. We apply a threshold of 0.5 for the multi-label task of SONYC-UST, and we consider the class with the highest output value as present for UrbanSound8K multi-class classification task. We found that our proposed method outperforms a manual aggregation method similar to the one proposed by [24], which gave poorer results on both datasets using our models.

5. RESULTS

Table 2 summarizes the performance of the methods on the TAU Urban Acoustic Scenes 2020 Mobile dataset. The parameter tuning procedure mentioned in section 4.3 identify a CNN model with a kernel size of 5, no dilation, 64 channels, and 5 layers, trained with a learning rate of 10^{-3}. This model contains 192,961 parameters, which represents 0.26% of PANN's and 5.2% of YamNet's total number of parameters. Our CNN-logits model outperforms the baseline models for PANN, achieving a PtoPa-FC of 89.3% and a lower KL-divergence than the baselines. When YAMNet is used as the target classifier, our CNN-logits model achieves a higher PtoPa-FC than the other models. Notably, the KL-divergence of our model is higher than that of the YAMNet-1/3-oct model. This suggests that while its predicted first class is closer to that of YAMNet, the overall distribution of predictions across all classes is further away from the ones of the pre-trained model.

The classification results of PANN and YAMNet models on the SONYC-UST and UrbanSound8k datasets using both original and transcoded Mel spectrograms as input are shown in Table 3. The state-of-the-art macro-AUPRC for a model that is fully trained on the SONYC-UST dataset is reported between 0.49 and 0.65 [22, 25]. In contrast, the best accuracy achieved on the UrbanSound8k dataset is 90% [26]. Despite not being specifically trained on these datasets, the PANN model using ground truth Mel spectrograms as input still achieves fairly good results, albeit not outperforming state-of-the-art models. PANN models that use transcoded Mel spectrograms as input have a 18.6% decrease in accuracy compared to when a ground truth Mel spectrogram is used. This is promising, as fast third-octave spectrograms contain much less information both on frequency and time dimensions. In contrast, using the transcoder for YAMNet resulted in a much more significant drop in accuracy.

Several factors may explain why the CNN-logits method performed less effectively when used with YAMNet. First, YAMNet is smaller and less accurate than the ResNet38 PANN model, as

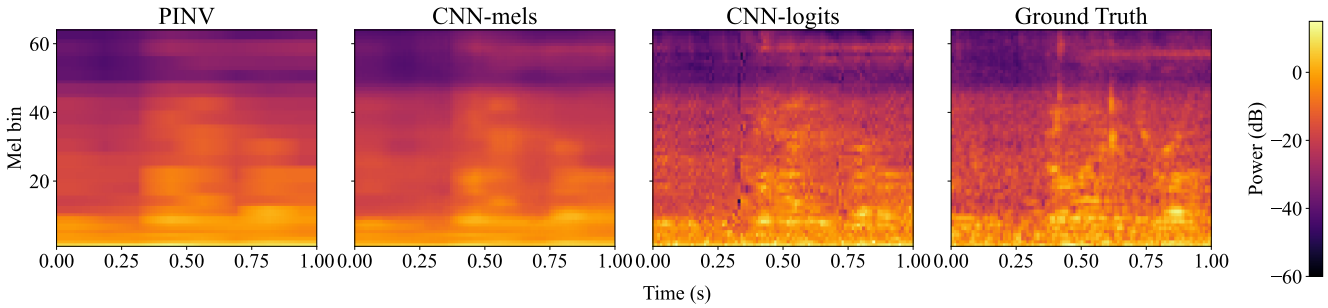

Figure 5: Mel spectrograms of a 1s file from the evaluation dataset, using different transcoding methods.

classifier	teacher-student	model	MSE (mels) ↓	KL divergence (logits) ↓	PtoPa-FC % ↑	training time
PANN	No	PINV	12.43	0.014438	0.4	-
		CNN-mels	**10.14**	0.013945	0.5	-
	Yes	CNN-logits	18.56	**0.000734**	**89.3**	14456s
		PANN 1/3 oct	-	0.000792	83.7	31702s
		Effnet-b0	-	0.008994	69.4	**11157**s
		Effnet-b7	-	0.006117	76.8	39653s
YAMNet	No	PINV	3.23	0.032255	0.8	-
		CNN-mels	**0.2**	0.013405	0.1	-
	Yes	CNN-logits	1.39	0.001863	**85.1**	5073s
		YAMNet 1/3 oct	-	**0.000919**	83.3	**4667s**
		Effnet-b0	-	0.005072	75.1	11153s
		Effnet-b7	-	0.003189	79.3	39701s

Table 2: Performance of the different models on TAU Urban Acoustic Scenes 2020 Mobile evaluation subset using pre-trained models predictions

input spectrogram	Mel		transcoded Mel	
classifier	PANN	YAMNet	PANN	YAMNet
accuracy on UrbanSound8k	**81.0** %	75.5 %	**62.4** %	42.7 %
mAUPRC on Sonyc-UST	**.52**	.48	**.44**	.34

Table 3: Performance of the different models on UrbanSound8k and Sonyc-UST. Bold values indicate the best scores achieved by classifiers using either ground truth Mel spectrograms as input (left), or Mel spectrograms transcoded from third-octave spectrograms.

evidenced by its lower performance on the multi-label and multi-class classification tasks in the UrbanSound8k and SONYC-UST datasets. Consequently, the coarser output logits of YAMNet compared to PANN suggest that the feature vector of size 521 produced by YAMNet may not be as relevant for spectrogram reconstruction. Additionally, YAMNet's Mel bins range from 125Hz to 7.5kHz, while PANN's Mel bins range from 20Hz to 14kHz, which is much closer to the range of third-octave bands (see Table 1). The 64 Mel bands of YamNet are confined within a narrower frequency range thus requesting for a reconstruction of higher frequency resolution, which could contribute to the enhanced difficulty in retrieving YAMNet's Mel bands from third-octave bands.

The CNN-logits method produces spectrograms that are more realistic and less blurry than those obtained using the CNN-mels and PINV baselines (as shown in Figure 5). This can be attributed to the fact that by minimizing the MSE between the two spectrograms, the algorithm tends to produce results that are closer to the ground truth in terms of average pixel-to-pixel distance but leads to globally blurry results. Conversely, by training on a set of 527 (or 521) high-level features, the neural network has more degrees of freedom and is not constrained to be as close to the ground truth spectrogram. As shown in Table 2, this is reflected in the lower MSE for the CNN-mels model than for the CNN-logits model.

6. CONCLUSION

In this study, we proposed a teacher-student approach to learning a transcoder whose task is to transform any spectral representation into a Mel spectrogram, for being used as input of pre-trained classifiers such as PANN and YAMNet models. This technique demonstrates a relatively high accuracy of 62.4% and macro AUPRC of 0.44 on UrbanSound8k and SONYC-UST, respectively, despite the limitations of a third-octave spectrogram in terms of temporal and frequency resolution.

However, one limitation of this method is that a new transcoder must be trained for each Mel spectral representation, in order to adapt to its different possible parameters (number of Mel bins, hop size, sample rate, etc...). To address this limitation, future research could explore reconstructing the audio entirely from a fast or slow third-octave spectral representation, which would allow the usage of any pre-trained classifier, including the state-of-the-art PANN model Wavegram-Logmel-CNN, which utilizes information on both time-domain waveforms and log Mel spectrograms.

Very interestingly, our experiments show empirically that predicted Mels using a loss built on logits do not only allow effective prediction but also results in Mels that have far better time / frequency structure.

7. REFERENCES

[1] Tensorflow, "Sound classification with yamnet," 2020, last access on 09/05/2023. [Online]. Available: https://github.com/tensorflow/models/tree/master/research/audioset/yamnet/

[2] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020, publisher: IEEE.

[3] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[4] IEC, "61672-1: 2013 Electroacoustics—Sound Level Meters—Part 1: Specifications," 2013.

[5] P. Aumond, A. Can, B. De Coensel, D. Botteldooren, C. Ribeiro, and C. Lavandier, "Modeling soundscape pleasantness using perceptual assessments and acoustic measurements along paths in urban context," *Acta Acustica united with Acustica*, vol. 103, no. 3, pp. 430–443, 2017, publisher: S. Hirzel Verlag.

[6] A. J. Torija, D. P. Ruiz, and A. F. Ramos-Ridao, "Application of a methodology for categorizing and differentiating urban soundscapes using acoustical descriptors and semantic-differential attributes," *The Journal of the Acoustical Society of America*, vol. 134, no. 1, pp. 791–802, 2013, publisher: Acoustical Society of America.

[7] M. Nilsson, D. Botteldooren, and B. De Coensel, "Acoustic indicators of soundscape quality and noise annoyance in outdoor urban areas," in *Proceedings of the 19th International Congress on Acoustics*, 2007.

[8] A. Can, J. Picaut, J. Ardouin, P. Crepeaux, E. Bocher, D. Ecotiere, and M. Lagrange, "CENSE Project: general overview," in *Euronoise 2021: European Congress on Noise Control Engineering*, 2021.

[9] F. Mietlicki, C. Mietlicki, and M. Sineau, "An Innovative Approach for long term environmental noise measurement: RUMEUR Network in the Paris Region," in *Proceedings of the EuroNoise*, 2015.

[10] J. C. Farrés, "Barcelona noise monitoring network," in *Proceedings of the EuroNoise*, 2015, pp. 218–220.

[11] F. Gontier, M. Lagrange, P. Aumond, A. Can, and C. Lavandier, "An efficient audio coding scheme for quantitative and qualitative large scale acoustic monitoring using the sensor grid approach," *Sensors*, vol. 17, no. 12, p. 2758, 2017, publisher: MDPI.

[12] F. Gontier, C. Lavandier, P. Aumond, M. Lagrange, and J.-F. Petiot, "Estimation of the perceived time of presence of sources in urban acoustic environments using deep learning techniques," *Acta Acustica united with Acustica*, vol. 105, no. 6, pp. 1053–1066, 2019, publisher: S. Hirzel Verlag.

[13] J. Hu, R. Ji, H. Liu, S. Zhang, C. Deng, and Q. Tian, "Towards Visual Feature Translation," in *Proc. IEEE Conf. Comput. Vis. Pat- tern Recognit.*, 2019, pp. 2999–3008.

[14] W. Kuang, Y.-L. Chan, S.-H. Tsang, and W.-C. Siu, "Fast HEVC to SCC transcoder by early CU partitioning termination and decision tree-based flexible mode decision for intra-frame coding," *IEEE Access*, vol. 7, pp. 8773–8788, 2019, publisher: IEEE.

[15] K. Zeng, J. Yu, R. Wang, C. Li, and D. Tao, "Coupled deep autoencoder for single image super-resolution," *IEEE transactions on cybernetics*, vol. 47, no. 1, pp. 27–37, 2015, publisher: IEEE.

[16] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *Proceedings of the European conference on computer vision (ECCV) workshops*, 2018, pp. 0–0.

[17] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, publisher: IEEE.

[18] M. Lagrange and F. Gontier, "Bandwidth extension of musical audio signals with no side information using dilated convolutional neural networks," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 801–805.

[19] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proc. Workshop Detection Classification Acoust. Scenes Events*, 2018.

[20] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*. PMLR, 2021, pp. 10 096–10 106.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] M. Cartwright, A. E. M. Mendez, J. Cramer, V. Lostanlen, G. Dove, H.-H. Wu, J. Salamon, O. Nov, and J. Bello, "SONYC urban sound tagging (SONYC-UST): A multilabel dataset from an urban acoustic sensor network," in *Proc. Workshop Detection Classification Acoust. Scenes Events*, 2019, pp. 35–39.

[23] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 1041–1044.

[24] F. Gontier, V. Lostanlen, M. Lagrange, N. Fortin, C. Lavandier, and J.-F. Petiot, "Polyphonic training set synthesis improves self-supervised urban sound classification," *The Journal of the Acoustical Society of America*, vol. 149, no. 6, pp. 4309–4326, 2021, publisher: Acoustical Society of America.

[25] A. Arnault and N. Riche, "CRNNs for Urban Sound Tagging with spatiotemporal context," *DCASE2020 Challenge, Tech. Rep.*, 2020.

[26] A. Gazneli, G. Zimerman, T. Ridnik, G. Sharir, and A. Noy, "End-to-end audio strikes back: Boosting augmentations towards an efficient audio classification network," *arXiv preprint arXiv:2204.11479*, 2022.