Full-Frequency Temporal Patching and Structured Masking for Enhanced Audio Classification

Aditya Makineni
Department of Computer Science
University of Alabama at Birmingham
Birmingham, AL, USA
amakinen@uab.edu

Baocheng Geng
Department of Computer Science
University of Alabama at Birmingham
Birmingham, AL, USA
bgeng@uab.edu

Qing Tian
Department of Computer Science
University of Alabama at Birmingham
Birmingham, AL, USA
qtian@uab.edu

***Abstract*—Transformers and State-Space Models (SSMs) have advanced audio classification by modeling spectrograms as sequences of patches. However, existing models such as the Audio Spectrogram Transformer (AST) and Audio Mamba (AuM) directly adopt square patching from computer vision, which disrupts continuous frequency patterns and produces an excessive number of patches, slowing training, and increasing computation. In this paper, we propose Full-Frequency Temporal Patching (*FFTP*), a patching strategy that better matches the time-frequency asymmetry of spectrograms by spanning full frequency bands with localized temporal context, preserving harmonic structure, and significantly reducing patch count and computation. We also introduce *SpecMask*, a patch-aligned spectrogram augmentation method that combines full-frequency and localized time-frequency masks under a fixed masking budget, improving temporal robustness while preserving spectral continuity. When applied on both AST and AuM, our method improves mAP by up to *6.76%* on AudioSet-18k and accuracy by up to *8.46%* on SpeechCommandsV2, while reducing computation by up to *83.26%*, demonstrating both performance and efficiency gains.**

***Index Terms*—audio classification, spectrogram representation, spectrogram patching strategy, model efficiency**

I. Introduction

Recent progress in deep learning for audio classification has been facilitated by architectures first introduced in other modalities, notably Transformers [1] and State Space Models (SSMs) [2], which have achieved state-of-the-art performance on large-scale benchmarks. Audio Spectrogram Transformer (AST) [3] and the more recent Audio Mamba (AuM) [4] adapt image-inspired architectures to audio classification by operating on log-mel spectrograms, a time–frequency representation of audio energy over time, with AST explicitly following the Vision Transformer (ViT) patch embedding paradigm [5] and AuM leveraging state space models for spectrogram sequence modeling. In AST, these patches are passed to a standard Transformer encoder, while AuM replaces the Transformer with the Mamba SSM architecture to enable long-context modeling with linear-time scaling. However, the square patching strategy in these models overlooks the asymmetric and anisotropic nature of spectrograms, where temporal events are localized (e.g., phonemes or acoustic events), while spectral structures such as harmonics, formants, and timbral signatures often span nearly the entire frequency axis and are best interpreted holistically. By using an identical resolution along both the temporal and frequency dimensions, square patching fragments these continuous spectral patterns across multiple tokens, forcing the model to reconstruct coherence through attention mechanisms or state transitions, which constitutes an inefficient inductive bias that may introduce unnecessary modeling errors. In addition to disrupting critical frequency continuity, the conventional square patch design also produces an excessive number of patches, increasing memory usage, training time, and computational cost without proportional performance gains.

To address these limitations, we propose Full-Frequency Temporal Patching (FFTP), a patching strategy tailored to the time-frequency characteristics of spectrograms. In our method, patches span the full frequency range while capturing localized temporal context. This preserves harmonic continuity and significantly reduces the number of patches compared to square patching. As a result, it improves the efficiency of both Transformer and SSM based architectures while aligning the model's receptive field with the natural structure of spectrograms. To further improve temporal robustness, we also introduce SpecMask, a patch-aligned spectrogram masking method tailored to FFTP. SpecMask combines full-frequency time masks with smaller, localized time-frequency masks under a fixed masking budget. By aligning augmentation masks to patch boundaries, SpecMask operates at the same granularity as the model's input tokens, enhancing regularization effectiveness while preserving spectral coherence and improving temporal robustness.

Experiments on two widely used audio classification benchmarks, AudioSet-18k [6] and SpeechCommandsV2 [7], show that FFTP combined with SpecMask consistently improves accuracy and mAP while significantly reducing computational cost.

II. Related Works

While square patches are commonly used for spectrogram processing with Transformers and SSMs, the use of alternative patch shapes has been little explored. The closest related studies investigate different kernel shapes for CNN-based spectrogram analysis. Pons *et al.* [8] improved music

* This work was partially supported by the National Science Foundation (NSF) under Award No. 2412285. (Corresponding author: Qing Tian.)

classification with vertical and horizontal CNN filters; Abdel-Hamid *et al.* [9] introduced limited weight sharing to create rectangular receptive fields that enhanced phoneme recognition; and Piczak [10] found rectangular filters more effective than square ones for environmental sound classification, particularly when aligned with the time axis. While these works highlight the benefits of anisotropic receptive fields, it is important to note that kernels in CNNs and patches in Transformers/SSMs are fundamentally different: CNN kernels slide locally across the input to capture local correlations, whereas Transformers/SSMs patches represent fixed regions that are embedded and processed as global tokens, without local sliding. In our work, we apply full-frequency temporal patches to Transformer and SSM architectures. Each patch serves as a global token in a sequence model, enabling long-range modeling across the entire frequency axis rather than local convolutional aggregation. Unlike prior approaches, we correct a mismatched inductive bias in spectrogram-based audio representations and redefine audio tokenization.

Meanwhile, spectrogram masking is essential for regularizing audio models. SpecAugment [11] applies random time and frequency masks, with variants for dynamic sizing [12] and mask scheduling [13]. However, these approaches ignore the patching structure of downstream models. This paper proposes SpecMask, which aligns masks with FFTP, combining full-band temporal masks with smaller localized time-frequency masks. This preserves spectral coherence while improving temporal robustness, making it well suited for patch-aligned architectures.

III. Methodology

Figure 1 provides an architecture overview of our framework. Details of our proposed FFTP and SpecMask will be discussed in Sec. III-A and Sec. III-B, respectively.

A. Full-Frequency Temporal Patching (FFTP)

In this paper, we propose Full-Frequency Temporal Patching (FFTP), a patching strategy that better aligns with the time-frequency asymmetry of audio spectrograms. Unlike conventional square patching, our method decouples the patch dimensions along the time and frequency axes, allowing each patch to span the full frequency range while capturing localized temporal context. Beyond improving efficiency, this patching method preserves harmonic and spectral structures, such as formants and harmonics, that extend across frequency bins.

Harmonics in audio signals generated by a single physical source (e.g., speech or musical instruments) appear as coherent stacks of energy spanning most of the frequency axis at a given time. Square patches fragment these stacks across multiple tokens, forcing the model to reconstruct spectral coherence. FFTP embeds each full-bank temporal slice as a single token, providing direct access to complete harmonic structures and enabling more accurate and sample-efficient learning. This results in semantically richer and more coherent token representations. The idea of FFTP is illustrated in Figure 2.

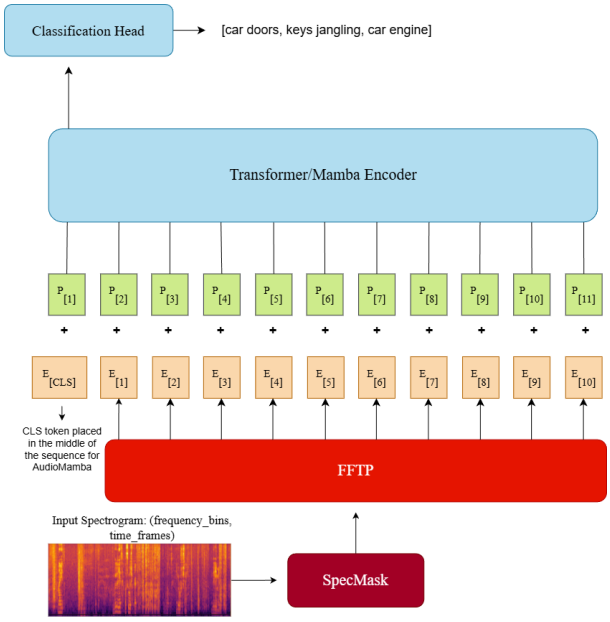


Fig. 1: Model framework overview. More details of FFTP can be found in Sec III-A and SpecMask in Sec III-B.

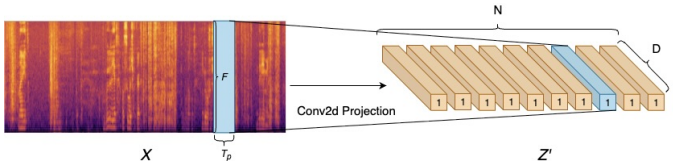


Fig. 2: Illustration of the proposed FFTP patching strategy. A log-mel spectrogram is projected into a sequence of D-dimensional embeddings using a 2D convolution layer with kernel size (F, T_p) and stride (F, s_t). Each patch spans the full frequency axis while capturing a short temporal window.

We begin with a log-mel spectrogram $X \in \mathbb{R}^{B\times 1\times F\times T}$, where B denotes the batch size, F the number of mel-frequency bins (e.g., 128), and T the number of time frames. FFTP partitions X into $N = [T/s_t]$ patches, each spanning the entire frequency axis and a temporal window of size T_p. Formally, the i-th patch is:

$$P_i = X[:, :, :, i \cdot s_t : i \cdot s_t + T_p] \in \mathbb{R}^{B\times 1\times F\times T_p},$$

where s_t is the temporal stride. To efficiently achieve this spectrogram partitioning, we apply a 2D convolution with kernel size (F, T_p) and stride $s = (s_f, s_t)$:

$$Z = \text{Conv2D}(X; W, s) \in \mathbb{R}^{B\times D\times 1\times N},$$

where $W \in R^{D\times 1\times F\times T_p}$ is the learnable convolutional kernel with full-frequency coverage, D is the embedding dimension, and

$$N = \left\lfloor \frac{T - T_p}{s_t} + 1 \right\rfloor$$

is the number of temporal patches. The output Z contains D-dimensional embeddings for each of the N time-localized

patches, with the original frequency dimension F projected into the embedding space. The result is then reshaped into a sequence of token embeddings:

$$Z' = \text{Transpose}(\text{Flatten}(Z)) \in \mathbb{R}^{B\times N\times D},$$

where each D-dimensional row represents the patch embedding of an audio sample at a specific time.

Instead of collapsing frequency information (e.g., via pooling), FFTP uses a learnable projection across frequency to extract knowledge; useful spectral information is preserved or enhanced via the D channel dimensions.

B. SpecMask: Patch-Aligned Spectrogram Masking

To improve the generalization of models under FFTP, we introduce *SpecMask* (Algorithm 1), a spectrogram masking strategy designed to align the masking structure with the time-frequency geometry of the input spectrogram.

Algorithm 1 SpecMask Algorithm

Input: $X \in \mathbb{R}^{F\times T}$, budget A, max size $(h_{\max}, w_{\max})$, mask type v
Output: X'
$M \leftarrow 0_{F\times T}, \quad a \leftarrow 0$
if $v =$ mean **then**
  $\mu \leftarrow \text{mean}(X)$
end if
while $a < A$ **do**
  if rand() < 0.7 **then**
    $h \leftarrow F,\ w \sim \mathcal{U}(1, w_{\max})$
  else
    $h \sim \mathcal{U}(1, h_{\max}),\ w \sim \mathcal{U}(1, w_{\max})$
  end if
  Sample (x, y) s.t. $M[x : x+h,\ y : y+w] = 0$
  Apply mask v to $X[x : x+h,\ y : y+w]$
  $M[x : x+h,\ y : y+w] \leftarrow 1, \quad a \leftarrow a + hw$
end while
return X'

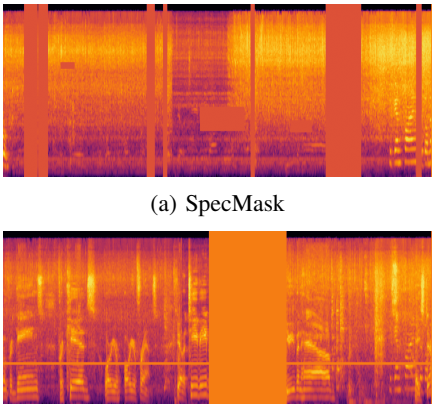
Unlike conventional masking methods such as SpecAugment, which apply time and frequency masks independently, SpecMask explicitly aligns mask shapes with FFTP's patch structure. As illustrated in Figure 3, SpecMask enforces a two-component masking strategy where, under a fixed masking budget (i.e., a fixed proportion of the time–frequency area), SpecMask applies:

1) **Full-band temporal masks** (e.g., 70% of budget): entire time frames (i.e., all frequency bins, or full columns) are masked, matching FFTP's full-frequency tokens.
2) **Localized time-frequency masks** (e.g., remaining 30% of budget): small rectangular regions within individual patches are masked, preserving patch diversity.

IV. Experiments and Results

A. Experimental Setup

To demonstrate the effectiveness of FFTP and SpecMask, we conducted experiments with two state-of-the-art audio models, AST and AuM, on two benchmark datasets: AudioSet (balanced subset) [6] and SpeechCommandsV2 [7]. Since different patching strategies cannot share the same pretrained checkpoint, for fairness, all models were trained from scratch without using external data, on a single NVIDIA A100 GPU with 80 GB of VRAM. The AudioSet-18k contains 527 multi-label audio classes, with each sample approximately 10 seconds long. Due to the unavailable or restricted content (e.g., deleted videos, private accounts), we gained access to 18,684 out of the original 22,176 samples through a pre-downloaded version made publicly available via Hugging Face [14]. SpeechCommandsV2 is a single-label dataset consisting of around 65,000 one-second utterances across 35 spoken words.



(a) SpecMask

(b) SpecAugment

Fig. 3: Visual comparison of the proposed SpecMask and standard SpecAugment masking strategies.

All audio was converted to mono and sampled at 16kHz. Spectrograms were generated with a resolution of *128×1000* for AudioSet (with mixup [15]) and *128×128* for SpeechCommandsV2, using the standard mel-filter bank settings from AST and AuM. SpecMask was used with a total area budget of 25,600 for AudioSet and 1,024 for SpeechCommandsV2. Models were trained for 25 epochs on AudioSet and 20 epochs on SpeechCommandsV2 using AdamW with cosine decay. All configurations converged stably within the specified number of epochs.

B. Quantitative Results

We evaluated performance using standard metrics such as mean average precision (mAP) for multi-label classification on AudioSet, and accuracy (Acc.) for single-label classification on SpeechCommandV2. As shown in Table I, FFTP consistently outperforms AST and AuM (with conventional square patches) across all datasets, and its performance is further boosted when combined with SpecMask. This advantage stems from the fact that, in a spectrogram, the time and frequency dimensions have distinct semantics and scales. FFTP can adapt patching to this asymmetry. By preserving the continuity of spectral patterns, FFTP introduces an inductive bias that aligns better with the

Model	AudioSet-18K (mAP)	Speech Comm. V2 (Acc.)
AST (Square)	11.25	85.27
AST with FFTP	**15.38**	**93.73**
AST with FFTP + SpecMask	**18.32**	**95.94**
AuM (Square)	13.28	91.58
AuM with FFTP	**14.24**	**94.68**
AuM with FFTP + SpecMask	**17.59**	**96.49**

TABLE I: From-scratch training results of AST and AuM, with and without FFTP and SpecMask. Models without "+ SpecMask" use SpecAugment, while "+ SpecMask" models use our proposed SpecMask. We train from scratch for fairness, since different patching strategies cannot share the same pretrained checkpoint. SpecAugment with FFTP was used with reduced frequency masking (max 5 bins) to avoid over corrupting the full-frequency tokens, which would otherwise erase entire harmonic structures.

Patch Shape	Patch Size	Stride	Patches
AST (Square)	(16, 16)	(10, 10)	1212
FFTP	(128, 50)	(128, 10)	96
	(128, 25)	(128, 5)	196
	(128, 10)	(128, 4)	248
	(128, 10)	(128, 2)	496
	(128, 10)	(128, 1)	991

TABLE II: Patch configurations used to train AST and our FFTP models along with the resulting patch counts for AudioSet18k.

underlying structure of audio signals (while also significantly reducing computation, as will be shown in Sec. IV-D).

C. Attention Overlay Analysis

We use Attention Rollout [16] to illustrate how the models attend to different regions of the spectrogram; Some representative examples are shown in Figure 4. Qualitatively, the proposed FFTP + SpecMask model focuses sharply on salient acoustic events (e.g., 'Keys Jangling' in the first example) while suppressing background noise. In contrast, the square-patch baseline produces diffuse attention, spreading focus across irrelevant regions. The precision of FFTP arises from its full-frequency patches, which preserve complete spectral harmonics, formants and foster more accurate temporal dependencies, along with richer contextual representations over time. As we will see next, FFTP also reduces fragmentation and greatly improves efficiency.

D. Efficiency Analysis

In modern deep models (particularly attention-based ones), patch count critically impacts efficiency. Table II shows the FFTP configurations we evaluated, all using full-frequency height ($F_p = 128$) but varying temporal window size T_p and stride s_t. The choice of temporal patch size T_p and stride s_t is guided by the duration of common acoustic events. While brief phonemes may last 10-50 ms, many real-world audio events (e.g., musical notes, environmental sounds) span 100-500 ms. We explore $T_p \in \{10, 25, 50\}$ frames (100 - 500 ms) and $s_t \in \{1, 2, 4, 5, 10\}$ frames to balance temporal resolution and overlap.

As shown in Table III, FFTP consistently outperforms square patching across all settings, achieving 18.32% mAP at 991 patches versus 11.25% at 1212 patches for square patching. Notably, even FFTP's most efficient variant (96 patches, 4.15 GFLOPS) exceeds square patching in both accuracy and speed. Despite using up to 991 patches, FFTP remains more computationally efficient than square patching. As shown in Table III and Figure 5, FFTP offers higher accuracy and efficiency, making it ideal for resource constrained and real-time audio applications.

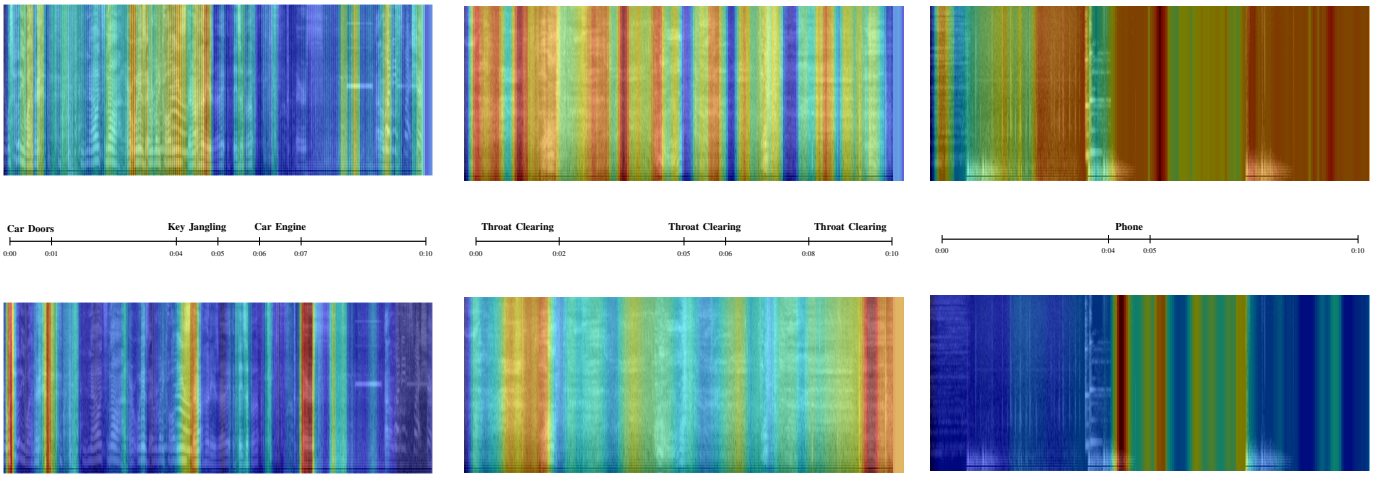
E. Ablation Studies

1) Component Isolation: To validate the individual contributions of our components, we perform a set of controlled experiments on AudioSet-18k by independently varying the patching strategy and masking method, as summarized in Table IV.

When using square patches, replacing SpecAugment (default in AST) with SpecMask results in a modest improvement from 11.25% to 12.04%, indicating limited benefit. In contrast, switching from square patching to FFTP while retaining SpecAugment yields a larger gain of +4.13% mAP, highlighting the representational advantage of frequency-aligned patches. Finally, combining FFTP with SpecMask leads to a substantial increase to 18.32% mAP, outperforming FFTP with SpecAugment by +2.94%. This combined effect suggests that SpecMask is explicitly complementary to FFTP's patch geometry, yielding larger gains when combined with FFTP than under square patching.

2) Performance Across Audio Event Types: To assess generalization across audio domains, we report mAP across four high-level audio event categories from the AudioSet-18k ontology: Human Voice (e.g., speech, laughter), Pets (e.g., dog, cat), Musical Instruments (e.g., keyboard, harp), and Tools (e.g., hammer, power tool).

As shown in Table V, FFTP + SpecMask consistently outperforms the square patching baseline with SpecAugment across all categories, with mAP gains ranging from 4.3% to 4.7%. Notably, the largest improvements occur in categories characterized by strong harmonic or formant structures such as human voice and musical instruments, where full-frequency coherence is most critcal. Even in less tonal categories like pets and tools, FFTP outperforms square patching, suggesting that temporal localization combined with spectral context benefits a broad spectrum of audio events. This category-wise analysis



(a) Sample 0RWZT-miFs (b) Sample -jcBFqcdpuI (c) Sample -5PZ_Bh-M6o

Fig. 4: Comparative attention visualizations across different audio samples. Top row: baseline AST attention maps. Middle row: ground-truth event timelines. Bottom row: proposed FFTP + SpecMask attention maps. Only the labeled regions in the timelines correspond to target sound events, while unlabeled regions represent background noise or non-target audio.

Patch Shape	Patches	GFLOPs	Train (hrs)	Inference (ms)	mAP (%)
AST (Square)	1212	103.35	5.3	14.50	11.25
FFTP + SpecMask	96	4.15	2.3	0.96	14.54
	196	17.30	2.5	2.18	17.15
	248	21.48	2.6	2.62	17.67
	496	42.79	2.8	5.38	18.01
	991	**85.32**	**5.2**	**11.62**	**18.32**

TABLE III: Model efficiency across patch counts on AudioSet-18k using AST. All measurements are performed on a single NVIDIA A100 GPU with a fixed audio input length of 10s.

Method	mAP (%)
Square Patch + SpecAugment	11.25
Square Patch + SpecMask	12.04
FFTP + SpecAugment	15.38
FFTP + SpecMask	18.32

TABLE IV: Data augmentation methods under different patching strategies.

Category	mAP (%)	
	Square + SpecAug	FFTP + SpecMask
Human Voice	12.4	16.8
Pets	8.9	13.2
Musical Instr.	10.7	15.4
Tools	7.3	12.1

TABLE V: Comparison of model performance across different audio event types.

confirms that FFTP's advantage is not limited to specific sound types but reflects a fundamental alignment with the time-frequency structure of audio.

3) Sensitivity Analysis of SpecMask Hyperparameters: We analyze the sensitivity of SpecMask to two key hyperparameters: total masking area budge A and full-band mask ratio r. Using AST with FFTP on AudioSet-18k, we vary $A \in \{10\%, 20\%, 30\%\}$ and $r \in \{0.5, 0.7, 0.9\}$.

Budget A	Full-Band Ratio r	mAP (%)
10%	0.7	16.34
20%		**18.32**
30%		17.27

TABLE VI: Sensitivity of SpecMask to masking budget A with a fixed full-band ratio r = 0.7.

Full-Band Ratio r	Budget A	mAP (%)
0.5	20%	17.21
0.7		**18.32**
0.9		16.92

TABLE VII: Sensitivity of SpecMask to full-band ratio r with a fixed masking budget A = 20%

The results in Table VI show that a moderate budget of 20% achieves optimal performance at 18.32% mAP, with lower and higher budgets yielding reduced performance. This suggests that a 20% masking budget provides an effective balance between preserving acoustic information and creating sufficient regularization.

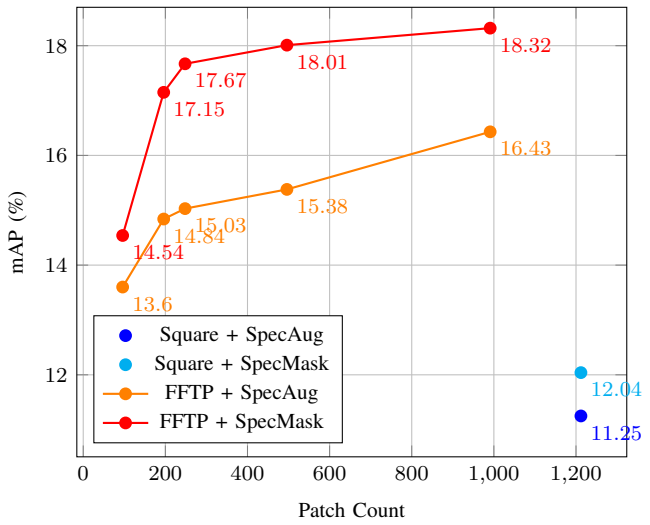


Fig. 5: mAP vs. patch count on AudioSet-18k for AST (Square patching + SpecAug), Square patching + SpecMask, FFTP + SpecAug, and FFTP + SpecMask. For fairness, we use no pre-trained checkpoints, as different patching strategies cannot share the same checkpoint.

Similarly, Table VII shows that a full-band ratio of $r = 0.7$ achieves the best performance, reaching 18.3% mAP, compared to $r = 0.5$ and $r = 0.9$. This indicates that a balanced combination of full-band and localized time-frequency masking is most beneficial for audio representation learning.

In general, while performance varies with the masking budget A and full-band ratio r, these variations are relatively small compared to the performance gaps between the proposed approach and square patching or SpecAugment, indicating that SpecMask is reasonably robust to hyperparameter choices.

V. Conclusion

In this paper, we propose Full-Frequency Temporal Patching (FFTP) and SpecMask, a patching and augmentation strategy tailored to the time-frequency asymmetry of audio spectrograms. Contrary to square patching borrowed from vision, FFTP preserves harmonic continuity by embedding full-band temporal slices, reducing patch count by up to 92% while improving mAP by up to 7.07%. SpecMask further enhances model robustness by aligning augmentation masks with FFTP's geometry, combining full-frequency temporal masks with localized time-frequency corruptions under a fixed budget. Together, these methods deliver consistent performance gains across both Transformer based (AST) and State-Space-based (AuM) architectures on AudioSet-18k and SpeechCommandsV2.

Our ablation and diagnostic analyses show that FFTP's benefits are not limited to specific sound types. Performance improves across diverse semantic groups, including human voice, pets, musical instruments, and tools, with the largest performance gains observed in classes with strong harmonic and formant structure. This finding suggests that FFTP aligns with the physical generative process of real-world audio, where sources emit energy across broad frequency bands.

Looking ahead, this work opens several promising future directions. A natural extension is multi-scale FFTP, which applies full-frequency patches at multiple temporal resolutions in parallel. Such a design could jointly model transient events (e.g., consonants or percussive sounds) and sustained tones (e.g., vowels or musical notes) within a single token sequence, potentially improving performance on complex polyphonic scenes. Finally, the core principle underlying FFTP is not limited to classification: full-frequency temporal tokenization may also benefit audio generation and related tasks, where preserving spectral coherence at each time step can lead to more natural-sounding outputs.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[2] A. Gu, K. Goel, and C. Ré, "Efficiently modeling long sequences with structured state spaces," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[3] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," in *Proceedings of Interspeech*, 2021, pp. 571–575.

[4] M. H. Erol, A. Senocak, J. Feng, and J. S. Chung, "Audio mamba: Bidirectional state space model for audio representation learning," *IEEE Signal Processing Letters*, vol. 31, pp. 2975–2979, 2024.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[6] 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)*, 2017, pp. 776–780.

[7] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[8] J. Pons and X. Serra, "Randomly weighted cnns for (music) audio classification," in *2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2019, pp. 336–340.

[9] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on audio, speech, and language processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[10] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th international workshop on machine learning for signal processing (MLSP)*, 2015, pp. 1–6.

[11] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proceedings of Interspeech*, 2019, pp. 2613–2617.

[12] H. Lu and B. Li, "Sample adaptive data augmentation with progressive scheduling," *arXiv preprint arXiv:2412.00415*, 2024.

[13] P. Byun and J.-H. Chang, "Effective masking shapes based robust data augmentation for acoustic scene classification," in *2023 8th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC)*, 2023, pp. 404–408.

[14] A. Keesing, "Audioset dataset on Hugging Face," 2023. [Online]. Available: https://huggingface.co/datasets/agkphysics/AudioSet

[15] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[16] S. Abnar and W. Zuidema, "Quantifying attention flow in transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, pp. 4190–4197.