Supplementary Materials for MAViL: Masked Audio-Video Learners

Anonymous Author(s)
Affiliation
Address
email

Supplementary Materials

The supplementary materials are organized as follows: In §A, we present the qualitative results of audio and video reconstruction. These results are obtained using the stage-1 MAViL's decoders, which are trained to reconstruct raw inputs. In §B, we offer the comprehensive experimental details and hyperparameter configurations for pre-training and fine-tuning on each dataset. In §C, we perform additional experiments to evaluate and analyze MAViL's performance. These experiments include:

1. Modality-wise masking ratio and masking type analysis.
2. Contrastive weights/ hyper-parameters analysis.
3. From-scratch and large model analysis.
4. Text-audio retrieval tasks on AudioCaps [1] and Clotho [2].

In §D, we discuss MAViL's societal impact and limitations.

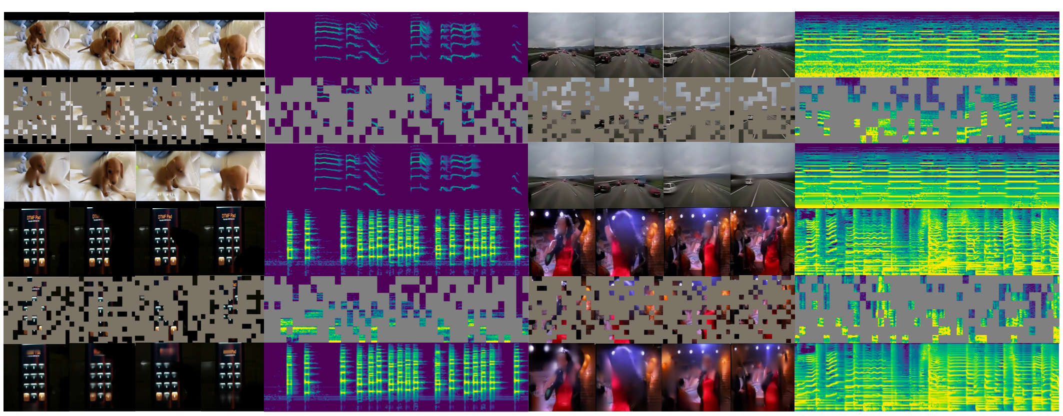

Figure 1: **Video clip and spectrogram reconstruction on the AudioSet *eval* set**. We sample 4 paired (video, audio) examples as follows: Top left: a puppy video; Top right: a recording from an ambulance's dash camera; Bottom left: a person dialing a phone in a dark room; Bottom right: a singer dancing. Input masking ratio: 70%. In each 3-row group, we show the original video and its audio spectrogram (top), masked input to MAViL (middle), and MAViL's video and audio spectrogram reconstructions (bottom). The spectrogram shape is 1024×128; patch size is 16×16. Each spectrogram has 64×8=512 patches. After applying 70% masking, there are 154 patches visible to MAViL. The 8-frame (4-second under 2 fps) video clip size is $8 \times 3 \times 224 \times 224$; patch size is 16×16. Each video has $4 \times 14 \times 14 = 784$ patches after patch embedding (temporal kernel/stride=2). After applying 70% masking, there are 235 patches visible to MAViL.

A Raw Audio-Video Reconstructions

In Fig. 1, we employ a stage-1 MAViL (ViT-B) to reconstruct raw audio spectrograms and video frames with masked inputs. The model is trained using an 80% masking ratio on the AudioSet-2M full training set with *un-normalized* raw spectrograms and video frames as the reconstruction targets (Eq.(4), stage-1). We visualize the reconstruction results by MAViL's audio and video decoders, wherein 70% of the input tokens are masked to its encoders. This visualization is performed on the AudioSet *eval* set.

The results demonstrate that MAViL effectively reconstructs highly corrupted versions of both audio spectrograms and video frames in video clips. The generated reconstructions for videos exhibit high fidelity and preserve spatial and temporal consistency of visual objects (*e.g*., the nearby moving cars recorded by the ambulance's dash camera) across different input domains, scenes, and lighting conditions. In the case of audio reconstructions, MAViL accurately maintains the positions and arrangements of time-frequency components in the spectrogram (*e.g*., the ambulance's siren and the song by the singer), which are essential for human understanding and perception of sound. Furthermore, the reconstructed audio and video components are consistent and well-aligned in time, enhancing the overall coherence of the reconstructed content.

B Experimental Details & Hyper-parameters

In this section, we provide additional experimental details for data preprocessing, implementation, pre-training, fine-tuning, and inference. The hyper-parameters are summarized in Table 1. The codebase and the pre-trained models will be available.

B.1 Data Preprocessing

In our study, we obtained a total of 2.01 million AudioSet videos, including both the video and audio tracks from the balanced and unbalanced training set and the evaluation set. Additionally, we managed to collect 198K VGGSound videos. As part of the preprocessing, we resized the video tracks to 360p while maintaining the aspect ratio and adjusting the longer dimension to 360 pixels. We also resampled the audio tracks to a sampling rate of 16K. We employed different temporal footprints for modeling the audio and video in MAViL, specified as the following:

Following the preprocessing in [3, 4, 5], we transform a raw audio (with mono-channel and under 16K sampling rate) into 128 Mel-frequency bands used in Kaldi [6]. This transformation involves using a 25ms Hanning window that shifted every 10ms. We then normalize the spectrogram according to the mean and variance in each dataset. For a 10-second audio, the resulting spectrogram has a dimension of 1024×128.

Regarding the video part, we utilize 4-second clips consisting of 8 frames captured at a rate of 2 frames per second (fps). Each input frame has a size of 224×224. In the pre-training phase, we apply common data augmentations such as random horizontal flip (with a probability of 0.5) and multi-scale random cropping (with a scale ranging from 0.2 to 1.0). In contrast, we apply only center cropping during the testing or inference phase. When processing a 10-second video clip from AudioSet, we randomly sample a starting point and extracted the consecutive 4 seconds of the video (cyclically looping back to the beginning if it was shorter than 4 seconds). As a result, the video clip input, consisting of 3 channels, had dimensions of $8 \times 3 \times 224 \times 224$.

B.2 Implementation

Uni-modal Encoders. We adopt the main design choices from original MAE for images [7] and Audio-MAE [5]. Specifically, we employ separate 12-layer Transformers with 12 attention heads as the encoders for each modality. The patch embedding and positional embeddings layers are also separated for each modality. During our investigation, we explored alternative designs, including sharing the audio-video encoder weights with separated inputs or concatenating them as done in Multi-MAE [8]. However, these alternative architectures resulted in inferior performance compared to the proposed architecture of using separated encoders with separated inputs. As a result, we chose to adhere to the original design of separate encoders for each modality.

In all Transformer encoders (with ViT-B as the default), the embedding dimension H is set to 768 For each input spectrogram of size 1024×128 representing a 10-second audio, we tokenize it into non-overlapping 16×16 spectrogram patches using an audio patch embedding layer. The kernel and stride sizes for both the time and frequency dimensions are 16, resulting in a total of 64×8 spectrogram patches or tokens for the audio sequence. The flattened audio token sequence has a length N of 512. Each audio token corresponds to a 768-dimensional vector. After appending the `[CLS]` token, adding positional embeddings, and applying 80% masking, the final input audio token sequence is represented as $\mathbf{a}' \in \mathbb{R}^{102\times768}$.

For each video clip with dimensions $8 \times 3 \times 224 \times 224$ (4 seconds in duration), we tokenize it into non-overlapping cells using a video patch embedding layer. The spatial kernel and stride sizes are set to 16, while the temporal kernel and stride sizes are set to 2. This process results in a total of $4 \times 14 \times 14 = 784$ video patches or tokens. The flattened video token sequence has a length M of 784. Each video token corresponds to a 768-dimensional vector. After appending the `[CLS]` token, adding positional embeddings, and applying 80% masking, the final input video token sequence is represented as $\mathbf{v}' \in \mathbb{R}^{156\times768}$.

Fusion Encoders. Following the ViT-B uni-modal encoders, we incorporate an audio-video *fusion* encoder. The fusion encoder consists of a two-layer (with L=2) Transformer, which can be either a vanilla Transformer or an MBT Transformer [3].

In the vanilla Transformer setup, the fusion encoder, denoted as $g_{\text{av}}(\cdot)$, jointly encodes the audio and video tokens. This is done by concatenating the output of the uni-modal encoders for audio ($\mathbf{a}\text{um}^{l+1}$) and video ($\mathbf{v}\text{um}^{l+1}$) as input, resulting in $(\mathbf{a}_{\text{um}}^{l+1} \| \mathbf{v}_{\text{um}}^{l+1}) = \text{Transformer}^l(\mathbf{a}_{\text{um}}^{l} \| \mathbf{v}_{\text{um}}{}^{l})$, where "$\|$" denotes concatenation.

In the MBT setup, we extend the vanilla Transformer by appending an additional 4 trainable MBT tokens for each modality. MBT encourages the model to more selectively collate and condense relevant information in each modality by forcing information exchange between modalities to pass through a small number of learnable bottleneck features $\mathbf{b}^0 = [b_1 \dots b_4], b_i \in \mathbb{R}^H$. The use of MBT tokens was originally proposed in the context of supervised audio-video learning. Precisely, $\mathbf{a}_{\text{um}}^{l+1} \| \mathbf{b}_{\text{a}}^{l+1} = g_{\text{av}}^l(\mathbf{a}_{\text{um}}^l \| \mathbf{b}^l)$ and $\mathbf{v}_{\text{um}}^{l+1} \| \mathbf{b}_{\text{v}}^{l+1} = g_{\text{av}}^l(\mathbf{v}_{\text{um}}^l \| \mathbf{b}^l)$, where $\mathbf{b}^{l+1} = (\mathbf{b}_{\text{a}}^{l+1} + \mathbf{b}_{\text{v}}^{l+1})/2$.

Decoders. The audio and video decoders are 8-layer Transformers with an embedding dimension of 512 and 16 attention heads. In the top decoder layer, we applied a linear prediction head to either predict the raw audio spectrogram and video frame patches in stage-1 (*i.e.*, $\mathbf{a}^{\text{raw}} \in \mathbb{R}^{H_{\text{raw}}^{\text{a}}}$ and $\mathbf{v}^{\text{raw}} \in \mathbb{R}^{H_{\text{raw}}^{\text{v}}}$), or predict the aliened and contextualized representations in stage-2 (*i.e.* $\mathbf{a}^{\text{Teacher}}, \mathbf{v}^{\text{Teacher}}, \tilde{\mathbf{a}}, \tilde{\mathbf{v}} \in \mathbb{R}^H$). The audio/video encoder and decoder in MAViL have 86M and 27M parameters, respectively. The floating point operations (FLOPs) for the audio encoder are 48.6G, comparable to the audio encoders in Audio-MAE [5] and CAV-MAE [9].

B.3 Training and Inference

Pre-training. MAViL operates under a fully self-supervised learning setup for pre-training. For pre-training MAViL's audio branch, we randomly initialize it from scratch. For the visual branch, we either randomly initialize it or initialize it with the self-supervised MAE [7] pre-trained on ImageNet where we simply repeat and inflate the convolution kernel in its patch-embedding to handle the additional temporal domain. Different visual initialization methods are compared in Table 6 in the main paper and Table 6 in Supplementary. Importantly, MAViL operates under the fully *self-supervised* setup.

MAViL is pre-trained on the combined unbalanced and balanced training sets of AS-2M. The pre-training process is performed using 64 GPUs with a 512 accumulated batch size. In stage-1 and each iteration of stage-2 (for $K = 3$ iterations), we pre-train the model for 20 epochs. Each pre-training session takes approximately 20 hours to complete. In total, the pre-training process takes around 80 hours. Note that the effective learning rate (lr_{eff}) depends on the base learning rate (lr_{base}) and the batch size. Precisely, $lr_{\text{eff}} = lr_{\text{base}} * \frac{\text{batch size}}{256}$. In our experiments, we also tried using strong data augmentations (*e.g.*, mixup [14], SpecAug [14], and CutMix [15]) to augment audio spectrograms during the pre-training phase. However, we observed that the resulting performance was either similar or worse compared to the baseline. Therefore, by default, we exclude these strong data augmentations for both audio and video during the pre-training phase.

Configuration	Pre-training AS-2M PT	Fine-tuning AS-2M	AS-20K	VGGSound	ESC	SPC
Optimizer	AdamW [10]					
Optimizer momentum	$\beta_1 = 0.9, \beta_2 = 0.95$					
Weight decay	0.00001					
Base learning rate	0.0002	0.0001†	0.001	0.0002	0.0005	0.001
Learning rate schedule	half-cycle cosine decay [11]					
Minimum learning rate	0.000001					
Gradient clipping	None					
Warm-up epochs	4	20	4	4	4	1
Epochs	20	100	60	60	60	10
Batch size	512	512	64	256	64	256
GPUs	64	64	8	32	4	4
Weighted sampling	False	True	False	True	False	False*
Weighted sampling size	-	200,000	-	200,000	-	-
Augmentation	R	R	R	R+N	R	R+N
SpecAug [12] (time/frequency)	-	192/48	192/48	192/48	96/24	48/48
Drop path [13]	0.0	0.1	0.1	0.1	0.1	0.1
Mixup [14]	0.0	0.5	0.5	0.5	0.0	0.5
Multilabel	n/a	True	True	False	False	False
Loss Function	MSE	BCE	BCE	BCE	CE	BCE
Dataset Mean for Normalization	-4.268	-4.268	-4.268	-5.189	-6.627	-6.702
Dataset Std for Normalization	4.569	4.569	4.569	3.260	5.359	5.448

Table 1: **Pre-training (PT) and Fine-tuning (FT) hyper-parameters**. For augmentation, R: sampling random starting points with cyclic rolling in time; N: adding random noise (signal-to-noise ratio (SNR): 20dB) to spectrograms. For loss functions, BCE: binary cross entropy loss (for multi-label datasets or when using mixup); CE: cross-entropy loss, MSE: mean square error loss. *: We repeat and balance each class to 50% of the size of the unknown class. †: For ViT-S, We use a learning rate of 0.0005 on AS-2M FT and 0.002 on AS-20K FT for the ViT-S model. For the ViT-L model, we use 0.0001 and 0.0005 for AS-2M and AS-20K FT experiments.

Fine-tuning. We fine-tune MAViL in three scenarios: (1) audio-only, (2) video-only, and (3) audio+video. We follow the setup in MAE and retain only the pre-trained uni-modal encoders for fine-tuning. In the audio-only and video-only setups, we fine-tune the respective encoders in the MAViL (stage-2). In the audio+video fusion setup, we introduce a 2-layer vanilla Transformer on top of the audio and video encoder in the MAViL (stage-2) and fine-tune it using both audio and video inputs. The hyperparameter configurations specified in Table 1 are employed for finetuning on each dataset. Empirically we observed a discrepancy in convergence rate between audio and video. We circumvent this by applying a 50% learning rate reduction for the weights of the video encoder when performing audio+video fusion fine-tuning.

We adopt the standard fine-tuning pipeline and augmentation in prior audio/audio-video classification works [4, 5, 3]. Specifically, we employ SpecAug [12], mixup [14], balanced sampling [16], and fine-tuning masking [5] (a 20% random masking rate for time and frequency in audio spectrograms; 20% for space and time in video clips). For video, we use standard video augmentations used in video classification [17, 18].

To perform importance sampling that balance the fine-tuning scheme on the unbalanced AS-2M (and VGGSound), we apply a distributed weighted sampler as prior works [16, 4, 19, 20]. We set the probability of sampling a sample proportional to the inverse frequency of its labels, where the label frequency is estimated over the training set. Specifically, for a instance i in a dataset $\mathcal{D}$ with a label pool $\mathbf{C}$, its sampling weight is proportional to $\sum_{c_i \in \mathbf{C}} w_c$, where $w_c = \frac{1000}{\sum_{i \in \mathbf{D}} c_i + \epsilon}$ and $\epsilon = 0.01$ is set to avoid underflow in majority classes. During the fine-tuning process on AS-2M, we randomly sample 200K instances (approximately 10% of AS-2M) with replacement in each epoch. We fine-tune MAViL for 100 epochs, which corresponds to approximately 10 full epochs of AS-2M. The entire fine-tuning process typically takes around 10 hours to complete.

Inference. After fine-tuning, we select the last checkpoint for inference. For the video and audio+video tasks, we adopt the standard approach used in video action recognition [21, 22, 23] by uniformly sampling ten 4-second video clips throughout the time domain of a video. Each of these sampled video clips is individually fed forward through the model to generate predictions. Note that for audio+video classification, the audio input remains the same 10-second audio recording throughout the sampling of video clips.

# Clips (AS-2M)	1	10
Audio	48.7	48.7
Video	29.4	30.3
Audio+Video	52.6	53.3

Table 2: **Number of video clips in the inference time.**

We average the ten predictions as the instance-level prediction and report the classification performance in Table 6 in §4. Note that these results are based on single-modal predictions, without ensembling multiple models. In Table 2, we compare the results obtained from one-clip predictions and ten-clip predictions (mAP on AS-2M). The sampling of ten clips leads to improvements of up to 0.9 mAP for video-only and audio+video tasks, while the audio-only task remains unaffected.

C Additional Experiments and Analysis

In this section, we present additional analysis to extend the study of the module-wise contribution in Table 3. We then expand our study on another important type of audio task: text-audio retrieval.

We organize this section as follows: Firstly, we investigate how different choices of masking ratio and masking type may affect the model performance. Next, we examine the effects of adjusting contrastive weights in the training objective. By exploring different weight settings, we aim to understand the influence of contrastive learning on the model's ability to capture audio-video relationships. Furthermore, we compare different approaches to visual backbone initialization and evaluate the performance using larger (ViT-L) audio/video encoders in MAViL-Large models. This analysis helps us understand the benefits and trade-offs of using larger backbone models and different initialization strategies. Additionally, besides audio-video classification tasks and audio-video retrieval tasks presented in the main paper. We include our study on audio-text retrieval tasks in the last.

Method	Audio	Video
A-MAE/V-MAE (baseline)	36.4	17.4
MAViL stage-1		
+ Joint AV-MAE	$36.8_{(+0.4)}$	$17.7_{(+0.3)}$
+ Intra and Inter contrast	$39.0_{(+2.2)}$	$22.2_{(+4.5)}$
MAViL stage-2		
+ Student-teacher learning	$41.8_{(+2.8)}$	$24.8_{(+2.6)}$

Table 3: **Module-wise Contribution** in MAViL).

C.1 Masking Ratio and Type

In addition to applying a shared masking ratio for each modality, we also investigated the impact of applying different masking ratios for audio and video. The results of this analysis are summarized in Table 4a. Interestingly, we did not observe a significant change in performance (mAP on AS-20K) when using different masking ratios for audio and video. Based on these findings, we simplify the approach by defaulting to an 80% masking ratio for both audio and video, as the Joint AV-MAE entry (the second row) in Table 3.

The default masking strategy in our model is random masking, which applies the same Bernoulli trial parameterized by a masking ratio (p) to each spectrogram or RGB patch. In Table 4b, we explored more advanced masking strategies and compare their impacts. For audio spectrogram, in addition to random masking (time-and-frequency agnostic with Bernoulli trials), we investigated time-masking (randomly masks multiple periods of time components) and frequency masking (randomly masks multiple frequency bands). We perform Bernoulli trials on time or frequency slots instead of

Ratio	70% (A)	80% (A)	90% (A)
70% (V)	36.7/17.5	36.8/17.5	36.4/17.3
80% (V)	36.7/17.2	36.8/17.7	36.8/17.4
90% (V)	36.5/17.3	36.6/17.6	36.8/17.5

(a) **Modality-wise Masking**

Type	70%	80%	90%
Random (A), Random (V)	36.7/17.5	36.8/17.7	36.8/17.5
Time-Freq (A), Random (V)	36.2/17.5	36.3/17.7	36.3/17.8
Random (A), Space-Time (V)	36.7/17.2	36.7/17.3	36.8/17.5
Time-Freq (A), Space-Time (V)	36.0/17.1	36.2/17.1	36.3/17.3

(b) **Masking Type**

Table 4: **Masking Ratio** and **Masking Type** (mAP on AS-20K).

individual patches. For video frames, we explored time-wise masking (randomly masking an entire frame) and space-wise masking (randomly masking a spatial patch across time). We set the masking ratio between spatial/frequency and time as 2:1 and adjusted the overall ratio from 70% to 90% for comparison with random masking.

Surprisingly, we do not observe improvements when applying these advanced masking strategies for multimodal pre-training. The simplest random masking approach achieved the best pre-training performance. This observation aligns with the findings in uni-modal MAEs [7, 18, 5], suggesting that the random masking strategy is effective and sufficient for multimodal pre-training.

C.2 Contrastive Weights

Table 5 showcases the impact of adjusting contrastive weights α and β in MAViL. The results show that fine-tuning these contrastive weights leads to improved performance. In our experiments, we set $\alpha = 0.1$ and $\beta = 0.01$ which yield the best performance.

It is important to note that the smaller contrastive weights in Eq.(4) do not imply that the contrastive objectives are less significant. The weights are chosen to scale and balance the gradients from the reconstruction and the two contrastive objectives to ensure they fall within a comparable range. This adjustment enhances training stability. Furthermore, the softmax temperatures used in NCE (Eq. (2)) are set as $\tau_c^{\text{inter}} = 0.1$ (more tolerant) for inter-modal contrastive learning and $\tau_c^{\text{intra}} = 1.0$ (stricter) for intra-modal contrastive learning. These temperature values help regulate convergence across modalities in the contrastive learning process.

α	0.3	0.1	0.05
Audio	41.5	41.8	41.4
Video	24.3	24.8	24.4

(a) **Inter-modal** α

β	0.1	0.05	0.01
Audio	41.3	41.5	41.8
Video	24.3	24.7	24.8

(b) **Intra-modal** β

Table 5: **Contrastive Weights** (mAP on AS-20K).

C.3 From-scratch Visual Backbone and Large Models

Under the fully self-supervised setup, MAViL initializes its audio branch from scratch and initialize its visual branch either from scratch or from a ImageNet self-supervised pre-trained MAE (IN-SSL). In this part, we further explore and compare the visual backbone initialization strategies under different model sizes.

As shown in the top two rows of Table 6, when considering MAViL-Base models, there is a small gap (-0.2 mAP on AS-20K) observed in the audio stream when discarding visual initialization from the ImageNet self-supervised model. However, a larger gap (-0.9 mAP) is observed in the video stream. A similar trend is observed in the AS-2M experiments. This discrepancy in the visual part can likely be attributed to biases and visual quality issues such as misalignment, title-only content, and low-resolution videos present in AudioSet.

To address this gap in the visual part, incorporating additional uni-modal pre-training steps could potentially improve model performance. For instance, conducting separate audio-only and video-only large-scale pre-training as the first step. In this work, we focus on audio-video pre-training solely on AudioSet for simplicity and for fair comparison with baselines. The possibility of incorporating additional pre-training steps is left for future research.

			AS-20K			AS-2M		
Model	A-init	V-init	A	V	A+V	A	V	A+V
MAViL-Base	scratch	IN-SSL	41.8	24.8	44.9	48.7	30.3	53.3
MAViL-Base	scratch	scratch	41.6	23.7	44.6	48.7	28.3	51.9
MAViL-Large	scratch	IN-SSL	42.1	27.1	45.3	48.8	32.4	53.3
MAViL-Large	scratch	scratch	42.3	25.3	45.1	49.1	30.6	52.5

Table 6: **Visual Backbone Initialization and Model Size** (mAP).

When using large models (ViT-L, rows 3-4), the gap in visual mAP (-1.8 mAP) still persists. Interestingly, the audio part of large models actually benefits from from-scratch visual initialization, showing an improvement of +0.2-0.3 mAP. Additionally, when comparing rows 1-2 to rows 2-3, the visual stream is benefited more by employing a larger (ViT-L) backbone. Across all the configurations (from-scratch or visual initialization with IN-SSL), MAViL consistently outperforms recent baselines (in Table 6 of the main paper) by a significant margin.

C.4 Text-Audio Tasks

Another important audio-centered multimodal application involves text-to-audio and audio-to-text retrieval tasks. In text-to-audio retrieval, the query is a text description, and the model performs a search over the (testing) audio collection by computing and ranking the similarity between the query embedding and the audio embeddings. To evaluate the audio representations learned by MAViL, following CLAP [24], we add a text encoder initialized from Roberta [25]. We perform fine-tuning with inter-modal contrast on the same training set used by CLAP. Specifically, AudioCaps [1] and Clotho [2], and LAION-630K [24]. In Table 7, we report recall@1, 5, and 10 on the testing sets.

	AudioCaps [1]						Clotho [2]					
	Text-to-Audio			Audio-to-Text			Text-to-Audio			Audio-to-Text		
Model	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
MMT* [26]	36.1	72.0	84.5	39.6	76.8	86.7	6.7	21.6	33.2	7.0	22.7	34.6
ML-ACT* [27]	33.9	69.7	82.6	39.4	72.0	83.9	14.4	36.6	49.9	16.2	37.6	50.2
CLAP [24]	32.7	68.0	81.2	43.9	77.7	87.6	15.6	38.6	52.3	23.7	48.9	59.9
MAViL	**37.3**	**72.8**	**84.5**	**49.3**	**81.8**	**91.5**	**17.2**	**41.0**	**53.5**	**23.3**	**49.5**	**63.6**

Table 7: **Text-to-Audio retrieval** and **Audio-to-Text retrieval** (R@1,5,10↑) on AudioCaps and Clotho. *: models trained without LAION-630K [24].

As shown above, MAViL significantly outperforms CLAP and other recent audio-text models, achieving new state-of-the-art performance on both audio-to-text and text-to-audio retrieval tasks. These results further validate the effectiveness of MAViL's representations not only in audio-video and audio-only tasks, but also in audio-text tasks.

D Limitations and Impacts

Limitations. There are several limitations associated with MAViL. Firstly, the scale of the data poses a limitation. The AudioSet [28] dataset used by MAViL, with two million samples, is approximately two orders of magnitude smaller than the text corpora used in recent language models [29, 25, 30]. It is also an order smaller than image corpora like ImageNet-21K used by MBT [3].

Another limitation pertains to the duration of each audio sample. The 10-second recording in AudioSet are relatively short, which can hinder the proper learning of distant temporal dependencies in audio and video. This limitation restricts the potential applicability of MAViL to tasks that require modeling longer audio sequences, such as automatic speech recognition (ASR). Regarding video modeling, due to GPU memory constraints and choice of video footprints, MAViL only models 4-second video segments. This limitation makes it challenging to effectively model long video sequences. Additionally, the presence of low-quality videos and misaligned audio-video pairs in AudioSet may adversely affects pre-training.

Potential Societal Impacts. The datasets used in this paper, including AudioSet and other end task datasets, were properly licensed and publicly available at the time of data collection. It is important to note that some of the data may have been removed by YouTube or the dataset uploaders. Most of the data in these datasets are licensed under the Creative Commons BY-NC-ND 4.0 license or the Creative Commons 3.0 International License.

To investigate the bias in AudioSet, we selected 200 videos containing speech. In these videos, we did not observe any visual bias in the sampled speakers, which encompassed a wide range of ages, races, and genders. However, it is possible that there may be biases in the distribution of population and ethnicity within AudioSet. It is important to exercise caution and be aware of the potential unintended gender, racial, and societal biases present in AudioSet, which serves as the pre-training data for MAViL.

Given that AudioSet consists of a vast collection YouTube videos, there is a potential risk that MAViL could learn to reconstruct sensitive personal information, which could then be exploited for malicious purposes, including the creation of audio deepfakes [31, 32]. To address this concern, the released MAViL would be discriminative models, specifically the audio and video encoders, rather than generative models such as decoders. This shift aims to mitigate the potential risks associated with generating synthetic content that could be misused.

References

[1] C. D. Kim, B. Kim, H. Lee, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *NAACL-HLT*, 2019.

[2] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: an audio captioning dataset," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 736–740.

[3] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, "Attention bottlenecks for multimodal fusion," in *NeurIPS*, 2021.

[4] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio Spectrogram Transformer," in *Interspeech*, 2021.

[5] P.-Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, C. Feichtenhofer *et al.*, "Masked autoencoders that listen," in *NeurIPS*, 2022.

[6] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[7] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *CVPR*, 2022.

[8] R. Bachmann, D. Mizrahi, A. Atanov, and A. Zamir, "MultiMAE: Multi-modal multi-task masked autoencoders," *arXiv preprint arXiv:2204.01678*, 2022.

[9] Y. Gong, A. Rouditchenko, A. H. Liu, D. Harwath, L. Karlinsky, H. Kuehne, and J. Glass, "Contrastive audio-visual masked autoencoder," *arXiv preprint arXiv:2210.07839*, 2022.

[10] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2019.

[11] ——, "SGDR: Stochastic gradient descent with warm restarts," in *ICLR*, 2017.

[12] 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," *ArXiv*, vol. abs/1904.08779, 2019.

[13] G. Larsson, M. Maire, and G. Shakhnarovich, "FractalNet: Ultra-deep neural networks without residuals," in *ICLR*, 2017.

[14] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR*, 2018.

[15] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *ICCV*, 2019.

[16] J. B. Li, S. Qu, P. Huang, and F. Metze, "AudioTagging Done Right: 2nd comparison of deep learning methods for environmental sound classification," *CoRR*, vol. abs/2203.13448, 2022.

[17] C. Wei, H. Fan, S. Xie, C. Wu, A. L. Yuille, and C. Feichtenhofer, "Masked feature prediction for self-supervised visual pre-training," *CoRR*, vol. abs/2112.09133, 2021.

[18] C. Feichtenhofer, H. Fan, Y. Li, and K. He, "Masked autoencoders as spatiotemporal learners," in *NeurIPS*, 2022.

[19] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *ICASSP*, 2022.

[20] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient training of audio transformers with patchout," *arXiv preprint arXiv:2110.05069*, 2021.

[21] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *ICCV*, 2019.

[22] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," in *ICCV*, 2021.

[23] Y. Li, C.-Y. Wu, H. Fan, K. Mangalam, B. Xiong, J. Malik, and C. Feichtenhofer, "Improved multiscale vision transformers for classification and detection," *arXiv preprint arXiv:2112.01526*, 2021.

[24] Y. Wu*, K. Chen*, T. Zhang*, Y. Hui*, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 2023.

[25] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019.

[26] A. Oncescu, A. S. Koepke, J. F. Henriques, Z. Akata, and S. Albanie, "Audio retrieval with natural language queries," in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*. ISCA, 2021, pp. 2411–2415.

[27] X. Mei, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, "Audio captioning transformer," in *Proceedings of the 6th Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, Barcelona, Spain, November 2021, pp. 211–215.

[28] 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 *ICASSP*, 2017.

[29] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019.

[30] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *NeurIPS*, 2020.

[31] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*. ISCA, 2015, pp. 2087–2091.

[32] A. Chintha, B. Thai, S. J. Sohrawardi, K. Bhatt, A. Hickerson, M. Wright, and R. Ptucha, "Recurrent convolutional structures for audio spoof and video deepfake detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 1024–1037, 2020.