Supplementary Material for Siamese Vision Transformers are Scalable Audio-visual Learners

Yan-Bo Lin and Gedas Bertasius

Department of Computer Science
University of North Carolina at Chapel Hill
{yblin,gedas}@cs.unc.edu

1 Implementation Details

For all of our experiments, the video length is set to 10 seconds. During the pretraining stage, we randomly select one video frame with a spatial resolution of 224×224 from all available video frames. For audio preprocessing, we first resample the audio waveform to 16 kHz and then compute the audio spectrogram using PyTorch's [4] kaldi fbank. This process includes 128 triangular mel-frequency bins and a frameshift of 10 milliseconds. A 10-second audio spectrogram has a spatial resolution of 1024×128. We follow the standard ViT pipeline, which first patchifies the data and then computes the self-attention mechanism for both audio and visual tokens. For the MAE loss, we adopt the same loss function as used in CAV-MAE [2], applying the stopping gradient before the multimodal layers. During fine-tuning, similar to CAV-MAE [2], we aggregate all predicted probabilities across all video frames (i.e., 10 video frames) to make more accurate predictions on video data. This approach may be less effective than the video encoders used in AV-MAE [1] and MAViL [3]. We include all the detailed hyper-parameter settings in Table 4, Table 5, and Table 6.

Table 1: Comparison of AVSiam with AV-MAE when both methods are trained from scratch. The results are reported on the VGGSound dataset.

Method	AVSiam	AV-MAE
V100 Hours (↓)	**475**	2854
Acc. (↑)	**64.4**	64.2

2 Additional Results

Comparison with Training from scratch. In Table 1, we compare AVSiam trained from scratch with AV-MAE [1]. Our method achieves slightly higher accuracy on VGGSound while requiring **6**× less time for pretraining (**475** vs. **2854** V100 GPU hours).

Table 2: Single-modality results of AVSiam-Large and MAViL-Stage2 on AudioSet and VGGSound. Both methods are finetuned using single-modality inputs.

Method	V100 Hours	AS-2M (mAP↑)		VGGSound (Acc.↑)	
		A	V	A	V
MAViL-Stage2	5120	**48.7**	30.3	60.8	50.9
AVSiam-Large	**310**	48.5	**32.0**	**61.2**	**51.9**

Single-modality Results. In Table 2, we also compare our model with MAViL-Stage2 when both methods are finetuned with audio-only and visual-only inputs on the AudioSet and VGGSound datasets. Our AVSiam achieves similar or better results as MAViL while requiring **16**× less training time (**310** vs. **5120** V100 GPU hours).

Table 3: Ablating multi-ratio masking scheme on other architectures.

Method	AS-20K	VGGSound
CAV [2]	38.5	64.1
+ MAE Loss	42.0	65.5
+ Multi-Ratio Masking	**42.8**	**66.4**

Does multi-ratio masking benefit other architectures? In Table 3, we ablate the MAE loss and joint multi-ratio masking scheme with the MAE loss on the commonly used CAV (contrastive audio-visual learning) architecture [2], which uses separate audio and visual encoders. As with our shared encoder architecture, we observe consistent gains in this case.

Table 4: Hyper-parameter for AVSiam-Base.

	AVSiam-Base			
	Pretraining	Finetuning		
Dataset	AS-2M	AS-20K	AS-2M	VGGSound
Optimizer	Adam, weight decay=5e-7, betas=(0.95, 0.999)			
Backbone learning rate	1e-4	1e-4	5e-6	5e-5
LR Classifier (× encoders)	-	100	50	10
LR decay start epoch	10	2	5	2
LR decay rate	0.5	0.75	0.75	0.75
LR decay step	5	1	1	1
Epochs	20	15	15	15
Batch size per GPU	96	8	32	32
GPUs	16 × A5000		4 × A5000	
Class Balance Sampling	No	No	Yes	Yes
Mixup	No	Yes	Yes	Yes
Random Time Shifting	Yes	Yes	Yes	Yes
Loss Function	CL + MAE	BCE	BCE	CE
Input Norm Mean	-5.081	-5.081	-5.081	-5.081
Input Norm STD	4.485	4.485	4.485	4.485

Table 5: Hyper-parameter for AVSiam-Large.

	AVSiam-Large			
	Pretraining	Finetuning		
Dataset	AS-2M	AS-20K	AS-2M	VGGSound
Optimizer	Adam, weight decay=5e-7, betas=(0.95, 0.999)			
Backbone learning rate	1e-4	5e-5	5e-6	5e-6
LR Classifier (× encoders)	-	100	50	50
LR decay start epoch	10	2	2	2
LR decay rate	0.5	0.75	0.5	0.5
LR decay step	5	1	1	1
Epochs	20	15	15	15
Batch size per GPU	48	8	16	16
GPUs	32 × A5000	4 × A5000	8 × A5000	
Class Balance Sampling	No	No	Yes	Yes
Mixup	No	Yes	Yes	Yes
Random Time Shifting	Yes	Yes	Yes	Yes
Loss Function	CL + MAE	BCE	BCE	CE
Input Norm Mean	-5.081	-5.081	-5.081	-5.081
Input Norm STD	4.485	4.485	4.485	4.485

Table 6: Hyper-parameter for AVSiam-Huge.

	AVSiam-Huge			
	Pretraining	Finetuning		
Dataset	AS-2M	AS-20K	AS-2M	VGGSound
Optimizer	Adam, weight decay=5e-7, betas=(0.95, 0.999)			
Backbone learning rate	1e-4	1e-5	5e-6	5e-6
LR Classifier (× encoders)	-	100	50	50
LR decay start epoch	10	5	2	2
LR decay rate	0.5	0.5	0.5	0.5
LR decay step	5	1	1	1
Epochs	20	15	15	15
Batch size per GPU	24	4		
GPUs	64 × V100	8 × A5000		
Class Balance Sampling	No	No	Yes	Yes
Mixup	No	Yes	Yes	Yes
Random Time Shifting	Yes	Yes	Yes	Yes
Loss Function	CL + MAE	BCE	BCE	CE
Input Norm Mean	-5.081	-5.081	-5.081	-5.081
Input Norm STD	4.485	4.485	4.485	4.485

References

1. Mariana-Iuliana Georgescu, Eduardo Fonseca, Radu Tudor Ionescu, Mario Lucic, Cordelia Schmid, and Anurag Arnab. Audiovisual masked autoencoders. In *ICCV*, 2023. 1
2. Yuan Gong, Andrew Rouditchenko, Alexander H Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. Contrastive audio-visual masked autoencoder. In *ICLR*, 2023. 1, 2
3. Po-Yao Huang, Vasu Sharma, Hu Xu, Chaitanya Ryali, Haoqi Fan, Yanghao Li, Shang-Wen Li, Gargi Ghosh, Jitendra Malik, and Christoph Feichtenhofer. Mavil: Masked audio-video learners. In *NeurIPS*, 2023. 1
4. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 1