Language-Guided Audio-Visual Source Separation via Trimodal Consistency Supplemental Material

Reuben Tan[1] Arijit Ray[1] Andrea Burns[1] Bryan A. Plummer[1] Justin Salamon [2]
Oriol Nieto[2] Bryan Russell[2] Kate Saenko[1,3]
[1]Boston University, [2]Adobe Research, [3]MIT-IBM Watson AI Lab, IBM Research
{rxtan, aburns4, array, bplum, saenko}@bu.edu, {salamon, onieto, brussell}@adobe.com

In this supplemental, we provide the following additional material to the main paper:

A. Latent caption extraction

We provide an illustration of our latent caption extraction operation (Section 3.1) in Figure 2 and a more detailed description of the entire operation. As mentioned earlier, we extract a latent caption from each unlabeled video to provide pseudo-language supervision. Given a video V, we begin by encoding its center frame using the CLIP visual encoder: $f^V_{\text{center}} = g^V(V_{\text{center}})$. Symmetrically, we seek to extract a language representation that corresponds to the encoded center frame semantically, described next.

The encoding function of the CLIP language transformer encoder g^L provides a mechanism that is amenable to searching for latent captions that already exist in its learnt vocabulary, which allows us to freeze its parameters and leverage its strong visual-semantic alignment with the vision modality. Instead of using the trained token embeddings, we introduce a learnable token parameter p and pass it into the language encoder g^L. We adopt the simple objective function of maximizing the cosine similarity between the center frame representation and the output of the language encoder, which allows us to update the weights of p through gradient back-propagation. We formulate the optimization operation mathematically as:

$$p^* = \arg\max_p sim\left(f^V_{center}, g^L(p)\right) \tag{1}$$

where $sim(x, y) = x^T y/(\|x\|\|y\|)$ and $||.||$ denotes the L_2 norm operator. We compute the final latent caption of the video as $C^* = g^L(p^*)$. The latent captions are used in our proposed alignment objectives to provide pseudo-language supervision. The search time for parameter p in Equation 1 is about $\sim$148 seconds per video on a RTX 2080 GPU for 5k iterations.

B. Extraction of spatiotemporal region representations from CLIP in Section 3.1

We begin by providing an overview of the 2D attention pooling layer in the CLIP Resnet visual encoders. By default, the CLIP visual encoder outputs a global visual representation for each input image. While we use the Resnet variants instead of the transformer-based architectures in CLIP, the former differs from the standard Resnet architecture in two ways. First, the CLIP variant contains three convolutional stems instead of one. Second, and more importantly, the CLIP Resnet variant also replaces the global average pooling (GAP) layer with a 2D self-attention operation, which contains the key, query and value projections. Next, we describe in more detail this self-attention layer and how we modify it for our task.

CLIP 2D attention pooling. We begin by extracting a set of spatial region representations from an input image $\mathcal{I}$ as:

$f^I = g^V(I) \in \mathbb{R}^{HW \times D}$, where H, W and D are the downsampled height, width and channel dimensions. Recall that a self-attention operation involves the use of keys, queries, and values. The CLIP model computes an average image representation as the query vector: $\overline{f}^I = \frac{1}{HW} \sum_{j=1}^{HW} f_j^I$, where f_j^I denotes the j-th row of f^I. Then, it computes a final representation for the entire image as follows:

$$\begin{aligned} K &= \overline{f}^I W_K \in \mathbb{R}^{1 \times D} \\ Q &= f^I W_Q \in \mathbb{R}^{HW \times D} \\ V &= f^I W_V \in \mathbb{R}^{HW \times D} \end{aligned} \tag{2}$$

where W_K, W_Q and W_V are the key, query and value projection matrices, respectively and W_K, W_Q and $W_V \in \mathbb{R}^{D \times D}$. Lastly, we compute the final contextualized image representation as:

$$f^I_{\text{global}} = W_L \left(V^\top \text{softmax} \left(\frac{(QK^\top)}{\sqrt{D}} \right) \right) \tag{3}$$

where W_L is the final language projection layer that maps the visual representations into the joint visual-semantic embedding space and $W_L \in \mathbb{R}^{D \times D}$.

Modified attention operation. Our Multiple Instance Learning formulation necessitates the presence of region representations in each input frame since we are predicting a spectrogram mask for each region. Additionally, we require these region representations to be well-aligned with the language modality such that a region should have a high similarity with the language query if its visual concept is semantically consistent with that of the query. Consequently, we extract a set of spatiotemporal region representations f^V_{conv} for our input video V with T frames. We encode the t-frame as: $f^V_{t,\text{conv}} = g^V(V_t) \in \mathbb{R}^{HW \times D}$. Finally, we compute the set of language-aligned spatiotemporal region representations by projecting them through the value and language projection layers as follows:

$$\begin{aligned} f^V_{\text{val}} &= W_V f^V_{\text{conv}} \\ f^V &= W_L f^V_{\text{val}} \end{aligned} \tag{4}$$

We pass this set of spatiotemporal region representations into our audio separation model $\mathcal{M}$ along with an input audio spectrogram to predict a mask.

C. Mix-and-separate training objective in Section 3.2

Given an input video V, we begin by using the CLIP visual encoder to extract a set of language-grounded spatiotemporal region representations $f^V \in \mathbb{R}^{T \times H \times W \times D}$. For the j-th spatiotemporal region, we tile its visual representation by the factor $H^A W^A$ and concatenate them with the audio bottleneck representations (Figure 1) along the channel dimension: $f_j^{AV} = concat(f^A, tile(f_j^V))$, where f_j^{AV} has the dimensions $\mathbb{R}^{H^A \times W^A \times 2D}$. We pass the concatenated representations into the decoder $\mathcal{D}$ consisting of a series of upsampling convolutional layers to generate a real-valued ratio mask: $\hat{M}_j = \mathcal{D}(f_j^{AV}) \in \mathbb{R}^{F \times N}$. To predict the separated audio source, each element of the mask is multiplied with the corresponding location in the input spectrogram: $\hat{A}_j^S = \hat{M}_j \odot A^S$, where $\odot$ denotes the Hadamard product. The mask is then applied to the input spectrogram to predict the audio component corresponding to the video: $\hat{A}_j^S = \hat{M}_j \odot A^S$.

To train the audio U-Net decoder $\mathcal{D}$ to predict spectrogram masks given fused audio-visual and audio-text representation inputs, we use the self-supervised "mix-and-separate" learning objective since we do not have ground-truth audio source annotations within each training video. Specifically, we synthetically combine the audio of multiple videos and the goal is to use the visual information within each video to separate its corresponding audio waveform. This objective allows us to compute ground-truth ratio spectrogram masks for training without annotations. Next, we describe the generation process of the ground-truth ratio masks for a pair of videos which is also commonly used in prior work [3, 9]; the same process is generalizable to any number of input videos. Given a pair of ground-truth audio spectrograms A_1^S and A_2^S, we compute their ratio masks as follows:

$$M_1 = \frac{A_1^S}{A_1^S + A_2^S} \quad \text{and} \quad M_2 = \frac{A_2^S}{A_1^S + A_2^S} \tag{5}$$

We adopt the mask prediction loss [1,3,9] to train the audio U-Net decoder $\mathcal{D}$ for audio separation. Given the pair of predicted masks $\hat{M}_1$ and $\hat{M}_2$, we compute the mask prediction loss as:

$$\mathcal{L}_{mask} = ||\hat{M}_1 - M_1||_1 + ||\hat{M}_2 - M_2||_1 \tag{6}$$

We note that it is also possible to compute the above-mentioned L1 regression loss using the ground-truth audio spectrograms but prior work [3, 9] has demonstrated it is more numerically stable to use the ratio masks for supervision.

D. Ablation experiments

Ablation over region MIL mask prediction vs video-level prediction. We evaluate the effectiveness of learning to perform source separation at the region level as compared to the video level in Table 1. To perform video-level spectrogram mask prediction, we adopt the same video aggregation function in Sound of Pixels [9], where the region representations are maxpooled over the channel dimension to compute a final video representation that is passed into the

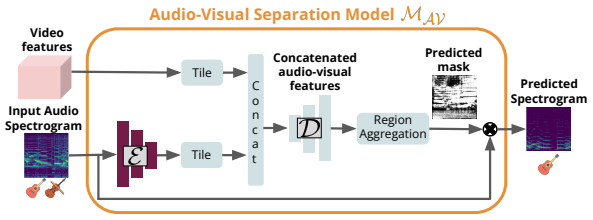

Figure 1. **Audio-visual separation approach in VAST.** We infer a predicted spectrogram mask for each spatiotemporal region and aggregate them to compute a final prediction for the input video.

audio U-Net decoder $\mathcal{D}$ (Figure 1). We note that our proposed alignment objectives are used in the training of both model variants. We observe that training a model to perform region-level predictions under the MIL formulation results in a significant performance gain over performing video-level predictions, which validates our hypothesis that a model trained to perform video-level predictions may not be able to identify candidate objects that emit sound.

Effect of sharing parameters in U-Net encoder $\mathcal{E}$. Prior work [3] learns a separate audio encoder for encoding the predicted audio waveforms to classify them according to discrete audio category labels. Here, we aim to determine the benefit of using shared parameters for our audio encoder component of the U-Net model $\mathcal{E}$ in Table 3. In this case, unlike prior work [3], we observe that using a shared audio U-Net encoder to encode the input audio spectrogram for source separation and the predicted spectrogram for the two new losses is integral to improving the final performance of our trained model on audio-visual separation.

Ablation over weights of $\mathcal{L}_{\text{Audio-language}}$ and $\mathcal{L}_{\text{Tri-modal}}$. We report the results of our ablation over the weights of our proposed audio-language and tri-modal consistency alignment objectives in Table 4. The results of adding the audio-language consistency loss seem to validate our initial hypothesis that using a lower weight term for this loss is beneficial. As discussed earlier in Section 3.1, this is similar to the multimodal contrastive formulation used for training joint vision-language foundation models such as CLIP and ALIGN. Thus, there is a high probability that we are treating some latent captions as false negatives for each video even though they may contain similar sounding objects. Setting a low weight helps to alleviate this negative consequence. However, we observe that the audio-language consistency loss is still very helpful for improving audio-visual source separation as well as learning a strong transitive alignment between the audio and natural language modality. The reported results also suggest that adding the tri-modal consistency loss also helps to improve performance significantly. In this case, we note that this alignment objective is formulated as a KL divergence minimization problem and does not require negative samples. Consequently, it may not be as important to use a low weight for this term as compared to the audio-language consistency objective.

Prediction	NSDR	SIR	SAR
Video-level	6.72	11.47	10.58
Region-level	**8.58**	**14.16**	**12.35**

Table 1. **Comparison between video-level and region-level audio predictions with our trained model on the SOLOS dataset.**

Replacing regions with bounding boxes. To determine if our approach can generalize well to pre-extracted bounding boxes during inference, we evaluate our trained model by replacing spatiotemporal region representations with those of bounding boxes during inference. We encode each bounding box as an image representation separately. Note that this is different from the region representations that are extracted from the modified self-attention operation in CLIP visual encoder (Section B). Consequently, our trained models may not generalize well to the different visual representations used during training and inference. We report our results in Table 2, where we observe that using bounding box representations in our trained models leads to a slightly lower performance in audio-visual separation.

Visualizations of latent captions. To understand what the latent captions encode, we provide some examples of their attention maps with respect to the video frames in Figure 3. Interestingly, we observe that a latent caption is capable of describing multiple instances of the same object in the middle visualization, where it is focusing on all three clarinets.

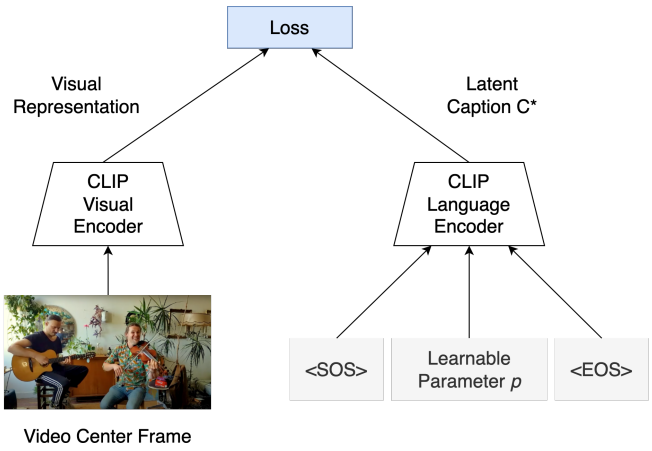

Figure 2. **Extraction of latent captions for pseudo-supervision.** We formulate the extraction mechanism as an optimization process and learn the weights of the parameter p by maximizing the cosine similarity between the final visual and language representations.

	NSDR ↑	SIR ↑	SAR ↑
Regions	8.58	14.16	12.35
Boxes	8.32	13.63	12.22

Table 2. **Evaluation on SOLOS.** We evaluate our trained model by replacing spatiotemporal region representations with those of detected bounding boxes and their representations.

E. Datasets

We train and evaluate our proposed VAST approach as well as other baselines on the widely-used SOLOS, MUSIC and AudioSet datasets which we describe below.

MUSIC [9]. The MUSIC dataset consists of videos that are downloaded from YouTube using queries about various musical instruments. It contains approximately 536 and 149 solo and duet videos, respectively. The entire set is comprised of videos containing 11 instrument categories: accordion, acoustic guitar, cello, clarinet, erhu, flute, saxophone, trumpet, tuba, violin and xylophone. Since the original splits of the dataset are not released, we adopt the same splits as [3], where the first and second videos in each instrument category are used as validation / test data and the rest are used for training.

SOLOS [5]. Similar to the MUSIC dataset, the SOLOS dataset contains 755 videos of musical videos that span 13 instrument categories. These videos are obtained from YouTube where the authors use queries of instruments as well as the 'solo' or 'auditions' tag. Unlike the MUSIC dataset, the SOLOS dataset does not contain duet videos.

AudioSet-Unlabeled [4]. AudioSet is a dataset that contains over two million 10 second video clips spanning 632 audio event classes that are sourced from YouTube. Compared to the MUSIC and SOLOS datasets, the audio clips in AudioSet are generally much noisier due to the presence of background sounds. Following prior work [3], we filter the video clips according to 15 musical instrument categories and select those from the 'unbalanced' split for training and the 'balanced' split for validation and testing.

F. Implementation details

We implement our proposed approach using the Pytorch deep learning library [6]. Consistent with prior work [3, 9], we downsample the audio clips to 11 kHz and use a Hann window size of 1022 samples[1] and a hop length of 256 samples in the STFT operation. This step results in an audio spectrogram of dimensions 512 x 256, which is re-sampled on a log-frequency scale to compute a final spectrogram of dimensions 256 x 256. We use the CLIP Resnet50 model [7] and its language encoder to extract a latent caption for each video as well as encode visual and language representations for audio separation. We set the dimension of the audio U-Net bottleneck features D to be the same as that of CLIP embedding space, which is 1024. We freeze the CLIP encoders during training and train the audio U-Net from scratch using a base learning rate of 4e-3. We train all models for 100 epochs with the SGD optimizer as well as using a linear warmup of 1000 steps and anneal the learning rate using a cosine decay schedule. We train our full model using 4 Quadro 6000 GPUs for approximately 8 days.

[1]While it is common to use powers of 2 as FFT size, we use 1022 as opposed to 1024 to be consistent with previous literature.

Figure 3. **Visual attention of latent captions.** We see that the latent captions tend to focus on salient foreground objects.

G. Limitations

While we have demonstrated that our proposed VAST approach is able to generalize well to free-form natural language queries for source separation, we observe that it is only able to handle visually descriptive adjectives such as *person playing a small trumpet* instead of *a loud trumpet*. We hypothesize that this limitation is due to a higher likelihood of visually descriptive adjectives appearing in the alt text of the pretraining dataset used by CLIP. Additionally, we only focus on separating sounds of different object classes. Our approach does not generalize well to discriminating between sounds from multiple instances of the same class (*cf*., Fig 5 middle showing that we can detect the clarinets but not distinguish the different instances). An example of such a challenging task is audio-visual speech separation, where there are two or more people speaking simultaneously and the goal is to separate for the speech for each person. Similar to existing audio-visual speech separation approaches [2, 8], future work can aim to address this limitation by leveraging representations of different instances and additional information in the form of object labels and speech narrations.

H. Demo video with predicted audio component generations

We provide a demo video where we evaluate our trained models on random videos in the wild which contain two instruments. The video contains 4 evaluation samples on

Shared audio encoder params	SOLOS			MUSIC			Audioset		
	NSDR ↑	SIR ↑	SAR ↑	NSDR ↑	SIR ↑	SAR ↑	NSDR ↑	SIR ↑	SAR ↑
No	7.52	12.68	10.22	7.39	13.25	9.81	3.27	6.48	11.51
Yes	**8.58**	**14.16**	**12.35**	**8.08**	**13.97**	**11.33**	**11.33**	**7.62**	**13.20**

Table 3. **Ablation over using shared parameters for audio U-Net encoder.** We observe that using a common audio encoder $\mathcal{E}$ to encode both mixed and predicted audio inputs for separation and localization, respectively, helps to improve performance on audio-visual separation.

$\mathcal{L}_{\text{Audio-language}}$ weight	$\mathcal{L}_{\text{Trimodal}}$ weight	NSDR ↑	SIR ↑	SAR ↑
0.0	0.0	5.47	10.55	10.95
1e-1	0.0	6.09	11.77	10.77
1e-2	0.0	8.08	13.74	12.18
1e-3	0.0	7.45	13.40	11.11
1.0	-	1.24	4.97	11.27
-	1e-1	8.02	13.82	11.76
0.0	1e-2	7.92	13.49	11.65
0.0	1e-3	8.10	13.84	11.79
0.0	1.0	6.81	12.61	11.00
1e-3	1e-2	8.58	14.16	12.35

Table 4. **Ablation results over the weights of the audio-language and tri-modal consistency alignment objectives on SOLOs.** We observe that the inclusion of the audio-language and tri-modal consistency alignment objectives is beneficial for audio-visual separation.

the task of audio-language source separation in the input videos. Additionally, we also localize the separated audio sources in the corresponding video frames. For the first task, our objective is to separate an audio input based on a natural language query and the goal of the second task is to localize the predicted separated audio in its corresponding video. Note that we use our full VAST model that is trained with our proposed audio-language and tri-modal consistency alignment objectives. For each evaluation sample, we provide the following in order:

1. Input video with mixed audio input (composed of two different instruments)
2. Separated audio predicted by the full VAST model of the first instrument
3. Attention heatmap between the first separated audio in (2) and the center frame
4. Separated audio predicted by the full VAST model of the second instrument
5. Attention heatmap between the second separated audio in (4) and the center frame

We observe that our full VAST model, that is trained without ground-truth text annotation or object bounding boxes, is generally able to separate the audio inputs based on natural language queries.

References

[1] Moitreya Chatterjee, Jonathan Le Roux, Narendra Ahuja, and Anoop Cherian. Visual scene graphs for audio source separation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1204–1213, 2021. 2

[2] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *arXiv preprint arXiv:1804.03619*, 2018. 4

[3] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3879–3888, 2019. 2, 3, 4

[4] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017. 4

[5] Juan F. Montesinos, Olga Slizovskaia, and Gloria Haro. Solos: A dataset for audio-visual music analysis. In *22st IEEE International Workshop on Multimedia Signal Processing, MMSP 2020, Tampere, Finland, September 21-24, 2020*. IEEE, 2020. 4

[6] 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. *Advances in neural information processing systems*, 32, 2019. 4

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 4

[8] Akam Rahimi, Triantafyllos Afouras, and Andrew Zisserman. Reading to listen at the cocktail party: Multi-modal speech separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10493–10502, 2022. 4

[9] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European conference on computer vision (ECCV)*, pages 570–586, 2018. 2, 4