Achieving Cross Modal Generalization with Multimodal Unified Representation Appendix

Anonymous Author(s)
Affiliation
Address
`email`

A Details of Downstream Tasks

Table 1: The illustration of all downstream tasks.

Task	**Pretrained Modality**	**Downstream Dataset**	**Generalization Direction**
Cross-Modal Event Classification	Audio-Visual Audio-Visual-Text	AVE	A⟶V V⟶A
Cross-Modal Event Localization	Audio-Visual Audio-Visual-Text	AVVP	A⟶V V⟶A
Cross-Modal Video Segmentation	Audio-Text Audio-Visual-Text	AVS-S4	A⟶T T⟶A

Cross Modal & Dataset Event Localization	Audio-Visual
	Audio-Visual-Text
		AVE & AVVP	A(AVE)⟶V(AVVP)
			V(AVE)⟶A(AVVP)

B Implementation and Training Details

B.1 Pretraining Details

Following previous work [1], we extract pool5 feature maps from sampled 16 RGB video frames by VGG-19 network for each 1s visual segment, then we use global average pooling over the 16 frames to generate visual feature maps as $7 \times 7 \times 512$-D. Also, we use the VGG-like model pre-trained on AudioSet to extract 128-D audio features for each 1s audio segment. For text representations, we first design several prompt templates for each video label of the VGGSound-AVEL [2] dataset, and transform it into a descriptive sentence, see Section. B.3 for more details. Then we use Bert [3] as our text encoder and can get 768-D feature for each word. For all modalities, we extract their semantic features dimension as 256-D. We use three convolution networks to get visual-specific feature as $T \times 2048$ and use an average pooling and Linear layer to compress it to apply mutual information minimization with the visual-semantic feature. Since audio and text do not contain spatial information, we use a Linear layer to get audio-specific and text-specific representations as $T \times 256$, respectively. For Cross-CPC, we set the default prediction steps as 2. For the CLUB applied in each modality, we divide its training process into first forward (mainly used to optimize the approximation network q_θ) and second forward (mainly used for MI minimization), during each epoch, the first forward will be updated 5 times and the second forward will be updated only once. We set the learning rate as 0.0004, the γ in MM-EMA as 0.99, and the batch size as 64. For the inactivated code reset mechanism, we set the N_{re} as 200.

For three modalities of unified representation pre-training, most details are the same as previous stage. We need to apply Cross-CPC between every two modalities, i.e., visual-text, visual-audio, and audio-text. And the default prediction step is set as 1. All the above pre-training experiments are conducted on one NVIDIA A100 GPU.

B.2 Downstream Tasks

Cross-Modal Event Classification: The AVE [1] dataset contains a total of 28 different event types, and the length of each audio and video is 10 seconds. Taking video-to-audio generalization (V2A) as an example: during downstream training, we use the Encoder obtained in the abovementioned pre-training phase to encode video input into the unified discrete space, the 10s second video can get discrete vectors of length 10. Then we use two Linear layers as the Decoder to map the discrete vectors into a 28-D feature space, and use the softmax function to find the event with the highest prediction probability and calculate the loss with Ground Truth. During the whole process, the parameters of the Encoder are frozen, only the Decoder will be updated. We set the learning rate as $2.5 \times e^{-4}$, and the batch size as 256. After training, we directly replace the video with audio as input, and use the obtained Encoder and Decoder to predict events in the audio. The process of audio-to-video generalization (A2V) is similar.

Cross-Modal Event Localization: The AVVP [4] dataset contains a total of 25 different event types, and the length of each audio and video is 10 seconds. Different from each video in the AVE dataset that only contains one event type, the audio and video in the AVVP dataset may contain multiple different event types. For example, in the same video, the audio information contains events A, B, while visual information contains events A, C, D. As with the previous task, we also use the pre-trained Encoder to map the video input to a unified discrete space, and then we use two layers of Linear as the Decoder to map the discrete vectors to a 25-dimensional space, activate with Sigmoid function and combine with Ground Truth to calculate the loss. Other settings are the same as the cross-modal event classification.

Cross-Modal Video Segmentation: The AVS-S4 [5] dataset contains 4932 five-seconds videos over 23 categories, including humans, animals, vehicles, and musical instruments. As the same with VGGSound-AVEL dataset, we transform the text label corresponding to each video into a descriptive sentence. Taking audio-to-text generalization (A2T) as an example: during downstream training, we use the Encoder obtained in Audio-Text or Audio-Visual-Text pre-training phase to encode audio input into the unified discrete space. Then we use the architecture proposed in AVS [5] as our visual encoder, audio-video interaction module, and video decoder. During training, the audio encoder is frozen while others are trainable. We use the Adam optimizer with a learning rate of $1e^{-4}$ and the batch size is set to 4. After training, we directly replace the audio with text as input to test the performance of text-based video segmentation. The process of text-to-audio generalization (T2A) is similar.

Cross Modal&Dataset Event Localization: To further prove that our model is widely applicable in various downstream tasks, we also test the cross-modal generalization ability in cross-dataset scenarios. There are 12 common event categories shared in AVE and AVVP datasets: **dog**, **car**, **helicopter**, **violin fiddle**, **frying food**, **motorcycle**, **acoustic guitar**, **banjo**, **baby cry**, **chainsaw**, **cat**, **accordion**. During the downstream task, we use the video or audio modality in the AVE training set to train the localization model in a weakly-supervised manner, and then directly test the event localization performance of the opposed modality in the AVVP val set. We choose the F1 score as the evaluation metric. The other training details are the same as the Cross-Modal Event Localization task. All the above downstream tasks are training and evaluating on one NVIDIA A100 GPU.

B.3 Text Prompts

We design different prompt templates for different event categories in VGGSound-AVEL and AVS-S4 datasets. For example, for event **race car**, we design the following prompts: *The roar of a high-speed race car engine.*; *The race car is running on the road and making a loud engine sound.*; *There is a high-speed race car running on the road.* For event **playing violin**, we can design the following prompts: *The sweet and melodious sound of a violin being played.*; *The player is playing the violin.*; *Someone is playing beautiful music on the violin.* For different kinds of events, we will design several unique templates for each event type according to their characteristics. Each piece of data will randomly choose a prompt from these templates as the audio-visual description.

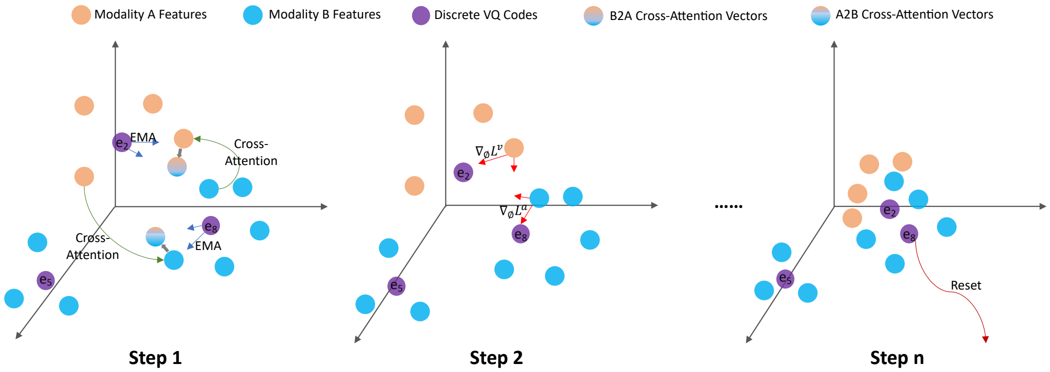


Figure 1: The overview of our proposed MM-EMA. Step 1, we use the cross-attention to obtain composite features and use EMA to update the latent VQ codes. Step 2, we use the modified commitment loss to update the representation of modality A and B. After several steps, these different modalities will be aggregated together, resulting in some redundant discrete code which will be reset.

Table 2: Ablation studies of audio-visual-text pre-training on three downstream tasks.

Method	VGGsounds-AVEL 40K							
	AVE		AVVP		AVE→AVVP		AVS-S4 (mIoU)	
	V→A	A→V	V→A	A→V	V→A	A→V	A→T	T→A
Encoder Frozen	**54.1**	**55.0**	**63.4**	**71.0**	**53.0**	**52.4**	**78.0**	**77.7**
Encoder Fine-tuned	44.5	54.9	54.0	50.8	44.5	50.0	77.8	77.1

C Details of Multi-Modal EMA

During MM-EMA, we use the cross-attention vectors $\mathbf{r}_i^b$ and $\mathbf{r}_i^a$ as the bridge to help the two modalities move closer to each other. At the initial training, the codes mapped from different modalities are not yet overlapped, thus the MM-EMA algorithm is similar to the original EMA equation:

$$N_i^{(t)} = \gamma N_i^{(t-1)} + (1-\gamma) n_i^{(t)} \quad \mathbf{o}_i^{(t)} = \gamma \mathbf{o}_i^{(t-1)} + (1-\gamma) \sum_{j=1}^{n_i^{(t)}} [\frac{\mathbf{z}_{i,j}^{a(t)} + \mathbf{r}_{i,j}^{b(t)}}{2}] \quad \mathbf{e}_i^{(t)} = \mathbf{o}_i^{(t)} / N_i^{(t)}, \tag{1}$$

with the help of the $\mathbf{r}_i^b$, $\mathbf{r}_i^a$ and the modified commitment loss, the distance between semantic features $\mathbf{z}_i^a$ and $\mathbf{z}_i^b$ is gradually decreasing. With the training progress, more and more latent discrete codes can be mapped from two modalities, thus the MM-EMA algorithm can be written as:

$$N_i^{(t)} = \gamma N_i^{(t-1)} + (1-\gamma)[n_i^{a(t)} + n_i^{b(t)}] \quad \mathbf{e}_i^{(t)} = \mathbf{o}_i^{(t)} / N_i^{(t)} \tag{2}$$

$$\mathbf{o}_i^{(t)} = \gamma \mathbf{o}_i^{(t-1)} + (1-\gamma) \Big[\sum_{j=1}^{n_i^{a(t)}} \frac{\mathbf{z}_{i,j}^{a(t)} + \mathbf{r}_{i,j}^{b(t)}}{2} + \sum_{j=1}^{n_i^{b(t)}} \frac{\mathbf{z}_{i,j}^{b(t)} + \mathbf{r}_{i,j}^{a(t)}}{2} \Big]$$

Finally, the information with the same semantics is converged into the same latent discrete code. This process will generate many redundant codes that will never be utilized, as shown in Fig. 1, thus, it is obliged to use the code reset mechanism to reset these quantized vectors.

D More Ablation Studies

D.1 The Effect of Whether Froze Encoder in Downstream Tasks

To evaluate whether the parameters of the pre-trained encoder should be frozen in downstream tasks, we conduct a series of experiments, as shown in Table. 2. The results demonstrate a significant decline in performance when the encoder is fine-tuned during these tasks. We argue the reason is that when the encoder parameters are not frozen, gradient updates from the downstream tasks may lead to updates in the encoder corresponding to the modality during training, thereby disrupting the alignment relationship between different modalities in the latent space.

Table 3: Ablation studies of the number of Cross-CPC prediction steps on three downstream tasks.

Prediction Steps	**VGGsounds-AVEL 40K**					
	AVE		AVVP		AVE→AVVP	
	V→A	A→V	V→A	A→V	V→A	A→V
step = 1	**54.1**	**55.0**	**63.4**	71.0	**53.0**	**52.4**
step = 2	51.8	54.4	58.2	**72.2**	50.1	48.2
step = 3	51.4	53.5	59.7	70.3	51.8	49.2
step = 4	52.7	51.8	50.0	62.9	47.4	40.1

Table 4: Ablation studies of Cross-CPC and InfoNCE on two downstream tasks.

Method	**VGGsounds-AVEL 40K**			
	AVE		AVVP	
	V→A	A→V	V→A	A→V
Cross-CPC	54.1	55.0	63.4	71.0
InfoNCE	54.2	55.9	46.5	39.7

D.2 The Effect of the Prediction Steps in Cross-CPC

In this paper, we set the default prediction step in the audio-visual pre-training experiments as 2, while setting it in the audio-visual-text and audio-text pre-training experiments as 1. To further illustrate the impact of the number of prediction steps on the effectiveness of unified representation, we conduct a series of experiments on audio-visual-text pre-training, as shown in Table. 3. Our findings reveal that when the prediction steps are less than 4, the model's performance in all downstream tasks exhibits only minor fluctuations. However, a significant decline in performance is observed when the prediction step reaches 4. This indicates that an excessive number of prediction steps can hinder the model's ability to learn fine-grained multimodal alignment relationships, as predicting the distant future often proves to be more challenging.

D.3 Replace Cross-CPC with InfoNCE

To further evaluate the effectiveness of our proposed Cross-CPC, we replace it with a traditional InfoNCE loss. We first compress the sequence audio, visual, and text information into single vectors and then directly apply contrastive learning among these three modalities. As we can see in Table. 4, when the Cross-CPC is replaced with InfoNCE, the model achieves comparable performance in cross-modal event classification tasks, but exhibits inferior performance in event localization tasks. The results demonstrate that in the case of coarse-grained alignment, our proposed Cross-CPC has similar effects to InfoNCE. However, in comparison to InfoNCE, our method attains fine-grained alignment of distinct modality information, which further demonstrates the effectiveness of our approach in multi-modal sequence unified representation.

E More Qualitative Analyses

Visualization of Latent Discrete Codes: In order to illustrate the process of our method aligning different modalities in the latent space more intuitively, we visualize the discrete latent codes obtained from different training epochs and compare our method with the baseline model. As shown in Fig. 2, we can see that there exist many inactivated codes in the baseline model in the early stage, with the audio and visual modalities converging separately rather than aligning. In contrast, our method effectively aligns these distinct modalities. The figure also showcases the efficacy of our inactivated code reset mechanism. In epoch 1, our approach leaves only a few codes unactivated, which are promptly reactivated in subsequent epochs. Meanwhile, the baseline model continues to exhibit a large number of inactivated discrete codes.

Visualization of Segmentation Results: Also we present more visualization results of A2T and T2A generalization of our model on AVS-S4 dataset. As we can see in Fig. 3 and 4, our model can accurately localize the area where the sound is produced even though the test modality has never been seen before.

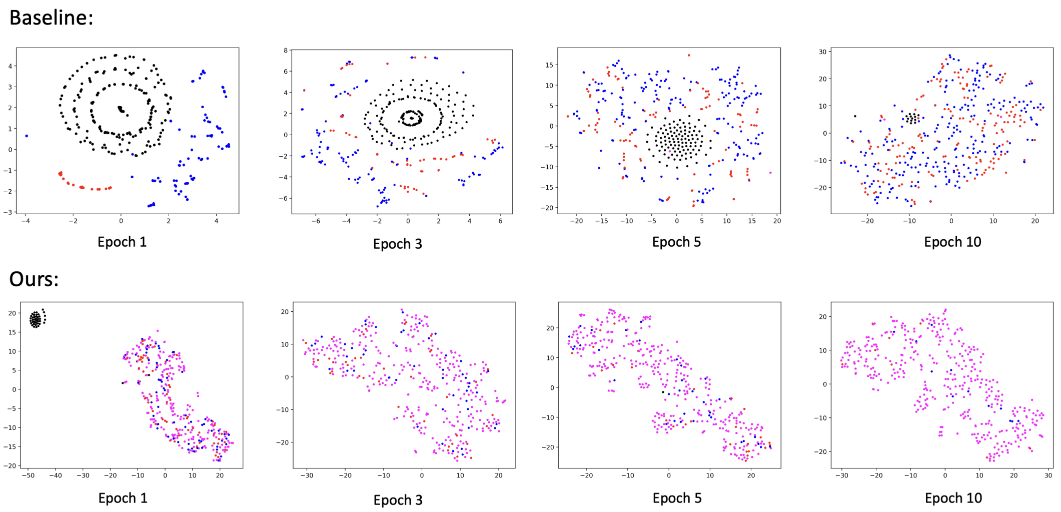


Figure 2: The visualization of latent codes of our model and the baseline model after training different epochs. Red represents audio-only code mapping, blue represents visual-only code mapping, purple represents audio-visual co-mapping, and black means that the code has not been activated.

Visualization of Fine-grained Prediction Score: To further illustrate that our model has fine-grained cross-modal generalization capabilities, we visualize the prediction results of the model on the AVVP dataset, as shown in Fig. 5. For different temporal segments, our model predict different probabilities for the occurrence of different events, and can accurately predict the types of events that appear in unknown modalities and the moments when these events occur.

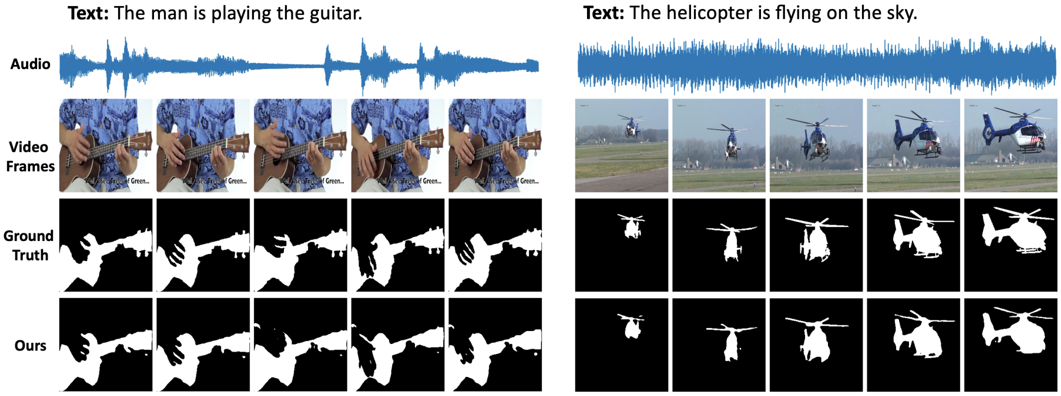

Text: The man is playing the guitar.
Text: The helicopter is flying on the sky.
Audio
Video Frames
Ground Truth
Ours

Figure 3: More visualization results of audio-to-text generalization on AVS-S4 dataset.

F Discussion

During the pre-training phase, our goal is to map information with the same semantics from different modalities together. To achieve this, we first utilize DCID to extract semantic information from various modalities, avoiding the introduction of modal-specific information that may affect alignment results. We then employ Cross-CPC for contrastive learning prediction, aligning different modalities at a fine-grained level. Subsequently, MM-EMA can assist in mapping these segments with identical information to the same discrete code. Once the training is complete, we can obtain the multi-modal unified representation. Then in the downstream training, we can assign additional knowledge for the known modality, such as event label, and relation with visual objects. Therefore, when the model is dealing with unknown modalities, by mapping it into a unified space, the corresponding labels learned from the known modality can be obtained, thus achieving cross-modal generalization and knowledge transferring.

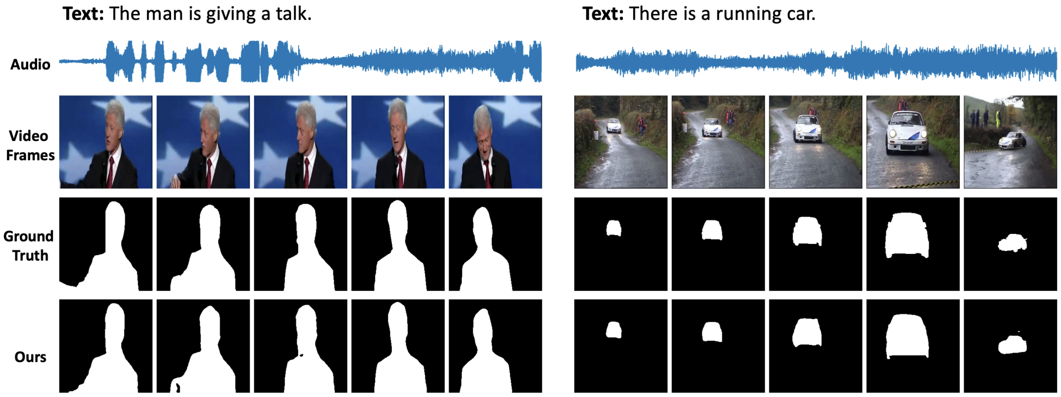
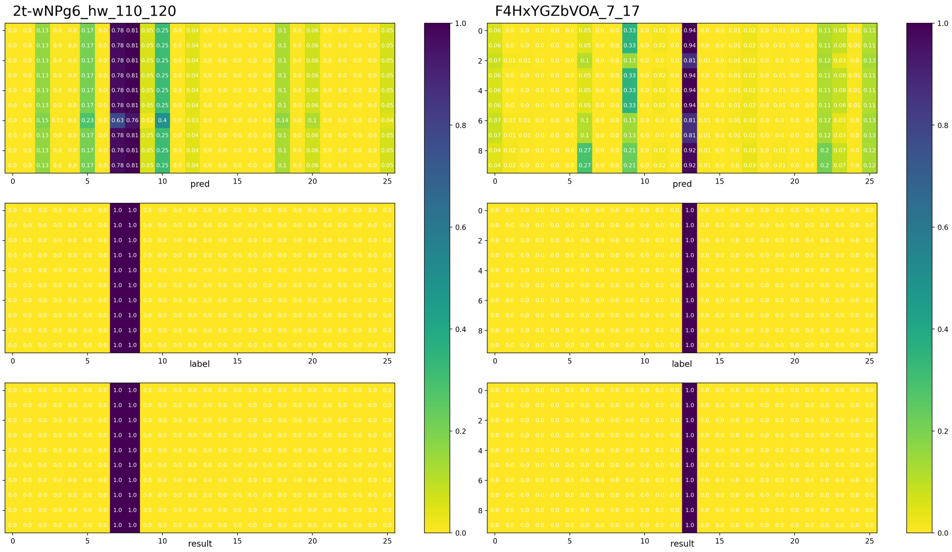

Text: The man is giving a talk.
Text: There is a running car.
Audio
Video Frames
Ground Truth
Ours

Figure 4: More visualization results of text-to-audio generalization on AVS-S4 dataset.



(a) video-to-audio generalization (b) audio-to-video generalizatoin

Figure 5: Visualization of the prediction scores on cross-modal event localization tasks. In each subfigure, the vertical axis represents temporal dimension, and the horizontal axis represents event types. The top layer is the possible score of each event at each segment predicted by the model. The middle layer is the ground truth label. The bottom layer represents the results of multiplying the prediction scores and ground truth.

References

[1] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 247–263, 2018.

[2] Jinxing Zhou, Dan Guo, and Meng Wang. Contrastive positive sample propagation along the audio-visual event line. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Yapeng Tian, Dingzeyu Li, and Chenliang Xu. Unified multisensory perception: Weakly-supervised audio-visual video parsing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 436–454. Springer, 2020.

[5] Jinxing Zhou, Jianyuan Wang, Jiayi Zhang, Weixuan Sun, Jing Zhang, Stan Birchfield, Dan Guo, Lingpeng Kong, Meng Wang, and Yiran Zhong. Audio–visual segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVII*, pages 386–403. Springer, 2022.