

# Patch-level Sounding Object Tracking for Audio-Visual Question Answering

Zhangbin Li\*, Jinxing Zhou\*, Jing Zhang, Shengeng Tang, Kun Li, Dan Guo†

School of Computer Science and Information Engineering, Hefei University of Technology  
lizhangbin.mail@gmail.com, zhoujxfut@gmail.com, guodan@hfut.edu.cn

## Abstract

Answering questions related to audio-visual scenes, *i.e.*, the AVQA task, is becoming increasingly popular. A critical challenge is accurately identifying and tracking sounding objects related to the question along the timeline. In this paper, we present a new Patch-level Sounding Object Tracking (PSOT) method. It begins with a Motion-driven Key Patch Tracking (M-KPT) module, which relies on visual motion information to identify salient visual patches with significant movements that are more likely to relate to sounding objects and questions. We measure the patch-wise motion intensity map between neighboring video frames and utilize it to construct and guide a motion-driven graph network. Meanwhile, we design a Sound-driven KPT (S-KPT) module to explicitly track sounding patches. This module also involves a graph network, with the adjacency matrix directly regularized by audio-visual correspondence map. The M-KPT and S-KPT modules are performed in parallel for each temporal segment, allowing balanced tracking of salient and sounding objects. Based on the tracked patches, we further propose a Question-driven KPT (Q-KPT) module to retain patches highly relevant to the question, ensuring the model focuses on the most informative clues. The audio-visual-question features are updated during the processing of these modules, which are then aggregated for final answer prediction. Extensive experiments on standard datasets demonstrate the effectiveness of our method, achieving competitive performance even compared to recent large-scale pretraining-based approaches.

## Introduction

Audio-Visual Question Answering (AVQA) is an attractive task in multimodal learning that involves the comprehension and utilization of audio, visual, and text modalities (Mao et al. 2024; Shen et al. 2023; Guo et al. 2024; Wang et al. 2024). As shown in Fig. 1(a), given a long audible video and a related question, the AVQA task requires predicting the correct answer, necessitating the accurate perception of the pivotal audio-visual signals that are relevant to the question. *Mainstream* methods on the AVQA task focus on designing efficient expert models. With limited training data, these

\*These authors contributed equally.

†Corresponding authors.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

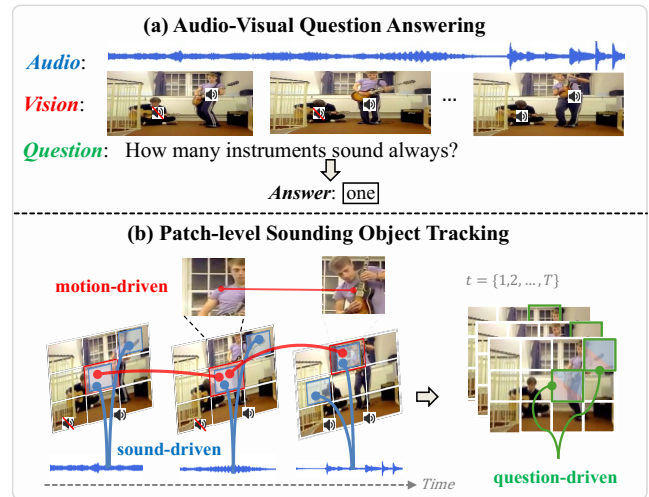


Figure 1: **Illustration of the AVQA task and our main idea.** (a) AVQA task requires accurate comprehension of sounding objects related to the question. (b) Our method explores the visual motion information and audio-visual correspondence to identify key salient and sounding patches. The question is then used to select the highly relevant patches.

methods emphasize the spatial-temporal reasoning of audio-visual scenes by exploring sounding regions (Li et al. 2022), question-related temporal segments (Jiang and Yin 2023; Li, Hou, and Hu 2023), or object-level cross-modal relations (Li et al. 2024). In particular, some methods attempt to utilize the frozen parameters of ViT (Dosovitskiy et al. 2020) models and insert learnable adapter layers to explore audio-visual relations, making the model suitable for the AVQA task (Lin et al. 2023; Duan et al. 2024). *Another branch* of methods relies on large-scale pre-training. For example, some works (Shu et al. 2023; Tang et al. 2024) attempt to extend large language models, *e.g.*, Vicuna (Touvron et al. 2023), into multimodal scenarios, making them capable of downstream AVQA task. Recently, VALOR (Chen et al. 2023) introduced a pretraining model that generates unified and advanced audio-visual-language representations, which generalizes well to AVQA task. However, all these methods require high computational resources or large-scale multi-

modal data during model pretraining/fine-tuning (we will provide more discussions and comparisons in the experiment section). Our work belongs to the mainstream branch, *i.e.*, building an expert AVQA model with sophisticated modules and a handful of training data.

Questions in the AVQA task usually involve some specific sounding objects. As the example shown in Fig. 1(a), the question “*How many instruments sound always?*” requires identifying the number of sounding instruments. However, accurately identifying and perceiving the sounding objects occurring in the video is not easy. **First**, the audio signal may be unclear, making it difficult to guide the identification of sounding objects. Moreover, the audio signal may fail to distinguish objects with identical appearances (same categories). For example, in Fig. 1(a), even if there is a clear sound of the guitar in the last segment, we cannot identify which guitar or whether both guitars emit the sound (instance-level identification). **Second**, the sounding objects can be changeable along the timeline. A sounding object may be salient in some temporal segments, while other salient objects may start to make sounds. This requires modeling/tracking the audio-visual correspondences for each temporal segment. **Third**, multiple objects may emit sounds. It is not enough to just perceive these sounding objects, but also to find the ones that are relevant to the given questions, *e.g.*, the guitar on the right “*sound always*” as queried in the question. According to these observations, we design meticulous modules and contribute a new **Patch-level Sounding Object Tracking (PSOT)** method for the AVQA problem.

**First**, to mitigate the potential deficiency of relying solely on audio signals, we propose to initially exploit the salient objects based on the visual information alone. Specifically, we identify salient visual patches with large movements, which are more likely to be sounding objects and helpful for question answering. As shown in Fig. 1(b), the man on the right is playing the guitar, and the relevant visual patches significantly change between adjacent frames. In contrast, the ceiling and the floor are constantly static, providing useless clues. To this end, we propose a **Motion-driven Key Patch Tracking (M-KPT)** module, which assesses the motion information between adjacent video frames by calculating the patch-wise feature similarity. The generated motion intensity matrix is used to activate the salient visual patches. Afterward, we construct a motion-driven graph network to propagate messages between features of visual patches and the synchronized audio, where the audiovisual nodes are guided by the motion intensity-driven adjacency relations. Concretely, for each temporal segment, the audio and original visual patches are treated as the initial *nodes* in the graph. Then we initialize the adjacency matrix (*i.e.*, *edge set*) by computing feature similarity between audio and *motion-enhanced* visual patches. In this way, the visual patches of salient objects are enhanced and tracked through video segments, and the audio feature updating also benefits from the tracked salient objects.

**Meanwhile**, to explicitly model the audio-visual correspondence between audio and visual nodes, we design a **Sound-driven Key Patch Tracking (S-KPT)** module. Sim-

ilar to the M-KPT module, the S-KPT module involves a graph network. These two modules share the same initial audiovisual nodes, and the primary difference lies in the adjacency matrix initialization. In the S-KPT module, we directly measure the audio-visual correspondence map to enhance sounding patches. Then, we generate the sound-driven adjacency matrix by computing feature similarity between audio and *sound-enhanced* visual patches. Notably, both audio and visual features are updated through the cross-modal interactions. The M-KPT module and S-KPT module are deployed in parallel, allowing the model to dynamically integrate the visual motion guidance or audio guidance.

The above two modules can identify salient and sounding visual patches. **Furthermore**, we design a **Question-driven Key Patch Tracking (Q-KPT)** module to identify and retain only those patches that are highly relevant to the question. This is also implemented in a graph network. We utilize the feature similarity (adjacency relations) between question and visual patches and retain the key patches with the highest pre-set percentage similarity values. For those visual patches not retained, their features are filled with zeros and would not be learned during graph learning. The Q-KPT ensures the model focuses more on the most informative visual patches related to the question. Given these preparations, we finally design a *Multimodal Message Aggregation* module to aggregate information from enhanced audio-visual-question features for answer prediction.

In summary, our main contributions are fourfold:

- By exploring patch-level clues of sounding objects, we present a novel graph-based PSOT model for the AVQA task. Compared with large-scale pretraining-based methods, our model achieves competitive performance.
- We propose the M-KPT module to identify salient visual patches with large movements. To the best of our knowledge, we are also the first to explore patch-level motion information in the AVQA task.
- We propose the S-KPT module to identify sounding patches. The S-KPT cooperates with the M-KPT, guaranteeing effective audio-visual representation learning via visual motion and audio guidance.
- We propose the Q-KPT module to further identify patches highly related to the question, enabling the model to utilize more informative information for answering.

## Related Work

**Audio-Visual Question Answering.** The AVQA task involves understanding audio-visual scenes (Zhou, Guo, and Wang 2023; Zhou et al. 2024b,c,a, 2022, 2024d, 2021, 2023; Guo et al. 2023) via a question-answering paradigm. To address the AVQA task, earlier works (Li et al. 2022; Li, Hou, and Hu 2023; Duan et al. 2024) emphasize spatial-temporal reasoning by perceiving sounding visual regions and question-related temporal segments. Recently, Li *et al.* (Li et al. 2024) utilize off-the-shelf object detection models to explicitly generate object-level visual features and then explore the relations among question, audio, and visual objects. Meanwhile, other works address the AVQA

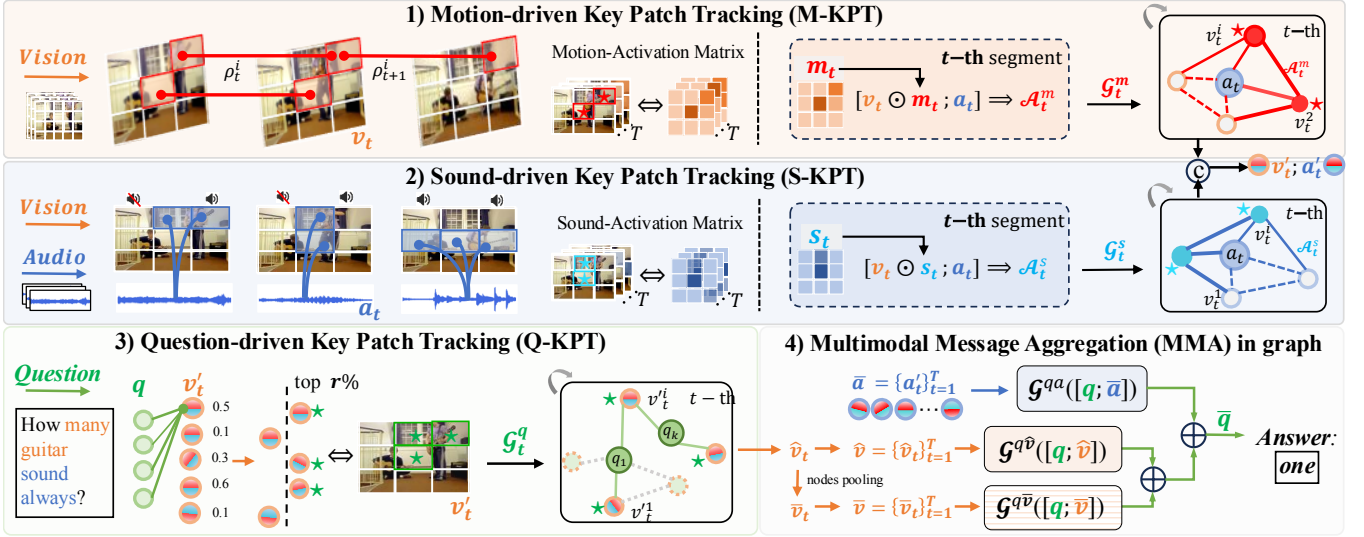


Figure 2: **Method Overview.** 1) The M-KPT module tracks salient visual patches ( $\star$ ) with large motion, which often relate to sounding objects and questions. It measures the patch-wise motion intensity information between neighboring frames, yielding the motion-activation matrix ( $m_t$ ) to guide adjacency matrix ( $\mathcal{A}_t^m$ ) for motion-driven graph network ( $\mathcal{G}_t^m$ ) learning. 2) Meanwhile, the S-KPT module tracks the sounding patches by assessing the audio-visual correspondence. A sound-driven graph network ( $\mathcal{G}_t^s$ ) is constructed. 3) The Q-KPT module further processes visual patches highlighted by the M-KPT and S-KPT modules, retaining only those patches highly relevant to the question. This is also achieved in a graph ( $\mathcal{G}_t^q$ ). 4) The MMA module integrates the question with enhanced audio&visual features through independent graph networks for answer prediction.

task by exploring data bias (Lao et al. 2023), missing modalities (Park, Oh, and Kim 2024), and large-scale data pre-training techniques (Shu et al. 2023; Tang et al. 2024). We solve the AVQA problem by exploring patch-level tracking of motion-aware, sound-aware, and question-aware patches. **Graph Neural Networks in QA tasks.** GNNs are flexible in encoding intra- and cross-modal relationships and facilitating message passing among nodes, making them popular backbones in traditional QA tasks such as Visual Question Answering (VQA) (Park, Lee, and Sohn 2021; De Oliveira Souza et al. 2023), Video Question Answering (Xiao et al. 2022, 2023), and text-based VQA (Hu et al. 2020; Biten et al. 2022). In the AVQA field, mainstream methods typically employ Transformer (Vaswani et al. 2017) blocks. Recently, some works have explored using the Mamba (Gu and Dao 2023) architecture to solve the AVQA problem (Huang et al. 2024; Yang and Li 2024). We implement our approach with the exploration of GNN, demonstrating competitive performance. Compared to traditional QA works, our graph-based method for AVQA has unique designs in constructing motion- and sound-driven adjacent matrices.

## Our Method

In the AVQA task, an audible video is first divided into  $T$  non-overlapping temporal segments. Each segment is represented by a 1-second audio signal and one visual frame. Let us denote the extracted preliminary audio and visual features as  $\mathbf{A} = \{a_t\}_{t=1}^T \in \mathbb{R}^{T \times d}$  and  $\mathbf{V} = \{v_t\}_{t=1}^T \in \mathbb{R}^{T \times N^2 \times d}$ , respectively. Here,  $d$  is the feature dimension, and  $N^2$  is the

number of patches (height  $\times$  width) in each video frame. As illustrated in Fig. 2, our framework for AVQA aims to identify the salient and sounding objects related to the question in each temporal segment and track them throughout the temporal sequences. We detail each proposed module next.

### Motion-driven Key Patch Tracking

As discussed in Introduction, the audio may be unclear and sometimes inefficient in identifying sounding objects sharing identical/similar appearances. To alleviate this issue, we consider first relying on visual information to highlight salient visual patches with large movements, which are more likely to be related to sounding objects and the given question. To this end, we propose a Motion-driven Key Patch Tracking (M-KPT) module.

Instead of using time-consuming optical flow (Horn and Schunck 1981), we calculate the discrepancy between adjacent frames at the patch-wise feature level to measure motion information. Specifically, for the  $i$ -th patch of the  $t$ -th video frame, whose feature is denoted as  $v_t^i \in \mathbb{R}^d$ , we compute its motion intensity  $\rho_t^i$  as follows:

$$\rho_t^i = 1 - \frac{v_t^i \cdot v_{t+1}^i}{\|v_t^i\| \|v_{t+1}^i\|}, \quad (1)$$

where  $v_{t+1}^i$  is the feature of adjacent  $t+1$  frame and  $\rho_t^i \in [0, 2]$ . If features of the  $i$ -th visual patch between adjacent frames have low cosine similarity, it will have a high  $\rho_t^i$ , indicating that the  $i$ -th patch may be part of a salient object. By performing Eq. 1 for each patch, we obtain the patch-wise motion intensity matrix for  $t$ -th frame, denoted as  $\rho_t$ .

$\rho_t = \{\rho_t^i\}_{i=1}^{N^2} \in \mathbb{R}^{N^2}$  reflects the motion intensity by capturing the patch-wise movement between adjacent video frames. Furthermore, in some scenarios, some salient or informative visual patches may only have large movements in some temporal frames. However, we believe that each video frame should emphasize these patches at the positions where the large movements exist in other frames. To track objects with large movement along the timeline, we additionally consider the global motion intensity across all temporal frames, computed as:  $\bar{\mu} = \frac{1}{T} \sum_{t=1}^T \rho_t$ . The total motion intensity matrix of  $t$ -th frame is a combination of the local motion matrix  $\rho_t$  and the global matrix  $\bar{\mu}$ :

$$\mathbf{m}_t = (1 - \lambda)\rho_t + \lambda\bar{\mu}, \quad (2)$$

where  $\mathbf{m}_t \in \mathbb{R}^{N^2}$  and  $\lambda$  is a constant, empirically set to 0.2.

After obtaining the motion intensity matrix  $\mathbf{m}_t$  for the  $t$ -th frame, we highlight those salient visual patches within the frame by multiplying  $\mathbf{m}_t$  with  $\mathbf{v}_t$ :  $\mathbf{v}_t^m = \mathbf{m}_t \odot \mathbf{v}_t$ , where  $\odot$  is the Hadamard product. Then, the motion-enhanced visual patch features  $\mathbf{v}_t^m \in \mathbb{R}^{N^2 \times d}$  are concatenated with the audio feature  $\mathbf{a}_t \in \mathbb{R}^{1 \times d}$ , which is then used to calculate the initial motion-driven adjacency matrix  $\mathcal{A}_t^m \in \mathbb{R}^{(N^2+1) \times (N^2+1)}$ . This process can be formulated as follows:

$$\mathcal{A}_t^m = [\mathbf{m}_t \odot \mathbf{v}_t; \mathbf{a}_t][\mathbf{m}_t \odot \mathbf{v}_t; \mathbf{a}_t]^\top, \quad (3)$$

where  $[\cdot; \cdot]$  represents the concatenation operator and  $\mathcal{A}_t^m$  reflects the pairwise relations.

The original visual features  $\mathbf{v}_t \in \mathbb{R}^{N^2 \times d}$  are concatenated with the audio feature  $\mathbf{a}_t \in \mathbb{R}^{1 \times d}$  as the basic nodes in the graph for  $t$ -th segment:  $\mathcal{N}_t = [\mathbf{v}_t; \mathbf{a}_t] \in \mathbb{R}^{(N^2+1) \times d}$ . With the nodes  $\mathcal{N}_t$  and adjacency matrix  $\mathcal{A}_t^m$  in hand, we construct a motion-driven graph neural network  $\mathcal{G}_t^m$  to encode the interactions between visual patches and audio features:

$$\mathcal{G}_t^m(\mathcal{N}_t) := \text{ReLU}(\mathcal{A}_t^m \mathcal{N}_t \mathbf{W}^m), \quad (4)$$

where  $\mathbf{W}^m \in \mathbb{R}^{d \times d}$  is a learnable parameter.  $\mathcal{N}_t$  is updated through  $L$  iterations, denoting  $L$  graph layers. We denote the audiovisual nodes after completing graph learning as  $\mathcal{N}_t^L$ . Notably, we construct the graph  $\mathcal{G}_t^m$  for each temporal segment, *i.e.*,  $t = 1, 2, \dots, T$ . The parameters are shared among different segments, ensuring computation efficiency. In this way, the salient visual patches are tracked along the temporal sequence, highlighted by the motion intensity matrix, and are then used to interact with corresponding audio node, benefiting effective audio-visual representation learning.

### Sound-driven Key Patch Tracking

The aforementioned M-KPT module relies on visual motion guidance and implicitly affects the audio node by interacting with motion-enhanced visual patches. To explicitly model the audio-visual correlations, *i.e.*, perceiving sounding patches, we propose the Sound-driven Key Patch Tracking (S-KPT) module.

Specifically, for the  $t$ -th segment, we directly assess the audio-visual correspondence between the audio node  $\mathbf{a}_t \in \mathbb{R}^{1 \times d}$  and visual patch nodes  $\mathbf{v}_t \in \mathbb{R}^{N^2 \times d}$ . The sound-activation matrix  $\mathbf{s}_t$  can be generated as follows:

$$\mathbf{s}_t = \frac{\mathbf{a}_t}{\|\mathbf{a}_t\|} \otimes \left( \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} \right)^\top, \quad (5)$$

where  $\otimes$  is the matrix multiplication and  $\mathbf{s}_t \in \mathbb{R}^{N^2}$ .

Similar to the M-KPT module, we further utilize the obtained  $\mathbf{s}_t$  to identify sounding patches within the frame by multiplying  $\mathbf{s}_t$  with  $\mathbf{v}_t$ . Then, the sound-enhanced features of visual patches are concatenated with the audio feature  $\mathbf{a}_t$  to compute the initial sound-driven adjacency matrix  $\mathcal{A}_t^s \in \mathbb{R}^{(N^2+1) \times (N^2+1)}$ :

$$\mathcal{A}_t^s = [\mathbf{s}_t \odot \mathbf{v}_t; \mathbf{a}_t][\mathbf{s}_t \odot \mathbf{v}_t; \mathbf{a}_t]^\top. \quad (6)$$

Given the audiovisual nodes  $\mathcal{N}_t = [\mathbf{v}_t; \mathbf{a}_t]$ , the sound-driven graph neural network can be formulated as,

$$\mathcal{G}_t^s(\mathcal{N}_t) := \text{ReLU}(\mathcal{A}_t^s \mathcal{N}_t \mathbf{W}^s), \quad (7)$$

where  $\mathbf{W}^s \in \mathbb{R}^{d \times d}$  is the learnable parameter. We perform  $\mathcal{G}_t^s$  for each temporal segment for the sounding patch tracking, and the parameters are shared across segments.

As shown in Fig. 2, we perform the M-KPT and S-KPT modules in parallel, enabling a joint and balanced utilization of the inherent visual motion and audio information to track salient and sounding patches within temporal segments. The updated audio-visual nodes output from the final layer of  $\mathcal{G}_t^m$  and  $\mathcal{G}_t^s$  are concatenated and sent to a fully connected layer for fusion. We denote the obtained audio and visual nodes as  $\mathbf{a}_t' \in \mathbb{R}^{1 \times d}$  and  $\mathbf{v}_t' \in \mathbb{R}^{N^2 \times d}$  ( $t = 1, 2, \dots, T$ ), respectively.

### Question-driven Key Patch Tracking

After highlighting the salient and sounding patches, we further consider that some of them may be redundant for answering the flexible questions. Therefore, we propose a Question-driven Key Patch Tracking (Q-KPT) module to retain only those patches within each frame highly relevant to the question, serving as refined clues for answering.

Given the word-level feature of the question  $\mathbf{q} \in \mathbb{R}^{K \times d}$  ( $K$  is the length of words) and the visual patches at  $t$ -th segment  $\mathbf{v}_t' \in \mathbb{R}^{N^2 \times d}$ , we first calculate their feature similarity ( $\mathbb{R}^{K \times N^2}$ ), similar to Eq. 5. Then, an average pooling operation along the word dimension is employed to obtain the sentence-level question-to-visual similarity, denoted as  $\alpha_t \in \mathbb{R}^{N^2}$ . Then, we propose to only reserve those visual patches that are among the top  $r$ (%) in terms of relevance to the question, as reflected in  $\alpha_t$ , for each frame. The mask matrix  $\beta_t \in \mathbb{R}^{N^2}$  indicating whether the  $i$ -th patch is retained can be computed as,

$$\beta_t^i = \begin{cases} 1, & \alpha_t^i \in \text{top } r \text{ of } \alpha_t; \\ 0, & \alpha_t^i \notin \text{top } r \text{ of } \alpha_t. \end{cases} \quad (8)$$

In this way, only  $N^2 * r$  visual patches are retained in each frame. For other patches (where  $\beta_t^i = 0$ ), their features are set to zeros in our implementation and will not be learned.

In fact, the values of  $\alpha_t$  and  $\beta_t$  can be computed and dynamically learned in a graph network. Specifically, the words of the question  $\mathbf{q} \in \mathbb{R}^{K \times d}$  and the visual patches  $\mathbf{v}_t' \in \mathbb{R}^{N^2 \times d}$  are concatenated as the basic nodes  $\mathcal{M}_t$  in a graph. We can calculate the initial adjacency matrix  $\mathcal{A}_t^q \in \mathbb{R}^{(K+N^2) \times (K+N^2)}$  as follows:

$$\mathcal{A}_t^q = \mathcal{M}_t \mathcal{M}_t^\top = [\mathbf{v}_t'; \mathbf{q}][\mathbf{v}_t'; \mathbf{q}]^\top. \quad (9)$$

The  $\alpha_t$  can be obtained from  $\mathcal{A}_t^q$  by utilizing only the adjacent weights between nodes of question words and visual patches. Accordingly, we can construct the graph neural network  $\mathcal{G}_t^q$ , formulated as,

$$\mathcal{G}_t^q(\mathcal{M}_t) := \text{ReLU}(\mathcal{A}_t^q \mathcal{M}_t \mathbf{W}^q), \quad (10)$$

where  $\mathbf{W}^q \in \mathbb{R}^{d \times d}$  is the learnable parameter. For convenience, we denote the updated features of visual patches at  $t$ -th segment as  $\hat{\mathbf{v}}_t \in \mathbb{R}^{N^2 \times d}$ . Notably, some visual patches have been discarded by setting their features to zeros. Similar to  $\mathcal{G}_t^m$  and  $\mathcal{G}_t^s$ ,  $\mathcal{G}_t^q$  is performed for each video frame for tracking the highly question-relevant patches along the timeline, and the parameters are shared across segments.

### Multimodal Message Aggregation

After going through the above introduced three modules, we obtain the enhanced audio and patch-level visual features, denoted as,  $\bar{\mathbf{a}} = \{\mathbf{a}'_t\}_{t=1}^T \in \mathbb{R}^{T \times d}$  and  $\hat{\mathbf{v}} = \{\hat{\mathbf{v}}_t\}_{t=1}^T \in \mathbb{R}^{T \times N^2 \times d}$ , respectively. For visual modality, the segment-level feature  $\bar{\mathbf{v}} \in \mathbb{R}^{T \times d}$  can be obtained from  $\hat{\mathbf{v}}$  via average pooling operation on the spatial dimension.

Given these outcomes, we propose the Multimodal Message Aggregation (MMA) module to comprehensively utilize the useful clues related to the question  $\mathbf{q}$  for final answer prediction. Here, we continue to adopt the graph network to model the interactions between audio/visual and question modalities. Specifically, the question feature  $\mathbf{q}$  is concatenated with the audio feature  $\bar{\mathbf{a}}$ , patch-level visual feature  $\hat{\mathbf{v}}$ , and segment-level visual feature  $\bar{\mathbf{v}}$ , respectively. Accordingly, we compute their feature similarity as the initial adjacency matrix. For simplicity, we abbreviate the constructed graph networks as  $\mathcal{G}^{qa}([\mathbf{q}; \bar{\mathbf{a}}])$ ,  $\mathcal{G}^{qv}([\mathbf{q}; \hat{\mathbf{v}}])$ , and  $\mathcal{G}^{q\bar{v}}([\mathbf{q}; \bar{\mathbf{v}}])$ , respectively. The question nodes are updated in each graph network. We then aggregate the learned features of the question nodes from the three graphs into a hidden feature  $\bar{\mathbf{q}} \in \mathbb{R}^{L \times d}$ , which is sent to a fully connected layer followed by a softmax function to generate the final probability  $\mathbf{p} \in \mathbb{R}^C$  for the candidate answer predictions. The cross-entropy loss is used for model training.

## Experiments

### Experimental Setup

**Dataset and evaluation metric.** We primarily conduct experiments on the widely-used and challenging MUSIC-AVQA (Li et al. 2022) dataset. This dataset consists of 9,288 videos involving various audio-visual music scenarios. Using 33 predefined QA templates, over 45,000 question-answer pairs are constructed. Most of the QA pairs are related to both audio and visual modalities (**AV-QA**), while the rest focus on either audio or visual modality alone (**A-QA/V-QA**). Following the standard protocol in the pioneering work (Li et al. 2022), we adopt the answer prediction accuracy (%) as the metric for model evaluation.

**Implementation details.** Each video in the dataset is 60 seconds long. To reduce computational costs, we sample one video frame every 6 seconds, resulting in  $T=10$  temporal segments. Accordingly, we obtain 10 1-second audio segments synchronized with the video frames. We employ the

image encoder of pretrained CLIP-ViT-L/14 (Radford et al. 2021) model to extract  $16 \times 16 \times 1024$ -D patch-level visual features, which are further downsampled to  $8 \times 8 \times 1024$  using average pooling to further reduce computational costs; For audio features, we use CLAP (Wu et al. 2023) to extract 512-D representations; For question embeddings, we utilize the text encoder of CLIP-ViT-L/14 to obtain 768-D word-level embeddings. We then apply linear projections to convert these features to the same 512-D. During model training, we use the AdamW optimizer with an initial learning rate of  $1e-4$ , which decays by 0.1 every 16 epochs. The batch size and epochs are set to 16 and 35, respectively. The numbers of graph layers in  $\mathcal{G}_t^m$ ,  $\mathcal{G}_t^s$ , and  $\mathcal{G}_t^q$  are empirically set to 3, 3, and 2, respectively. All experiments are conducted on an NVIDIA A40 GPU.

### Comparison with Prior Works

**Comparison with mainstream methods.** Our method belongs to the mainstream branch of AVQA works, focusing on designing expert models with limited training data. In Table 1, we compare our method with existing mainstream methods. **1)** Early works on the AVQA task typically utilize basic ResNet-18 to extract visual features. Recent works have started employing more powerful visual encoders, such as CLIP<sub>B</sub> (CLIP-ViT-B/32), and Swin<sub>L</sub> (SwinT-V2-L (Liu et al. 2021)). Our method PSOT<sub>B</sub> employs CLIP<sub>B</sub>/VGGish as the visual/audio feature extractor, the same as PSTP (Li, Hou, and Hu 2023), and surpasses PSTP by 1.77% in average QA accuracy. Notably, our method also slightly outperforms the previous state-of-the-art method CMP (Duan et al. 2024), which utilizes more advanced visual and audio encoders. These results demonstrate the effectiveness and superiority of our model design. Our model’s performance is significantly boosted when adopting more powerful CLIP<sub>L</sub> and CLAP models for feature extraction. The average performance reaches 78.42%. Our subsequent experiments are based on this configuration. **2)** Most mainstream works are implemented using Transformer (Vaswani et al. 2017). Some recent methods exploit the Mamba architectures and achieve competitive performance, *e.g.*, 74.12% accuracy achieved by SH-Mamba (Yang and Li 2024). Compared to these methods, our graph-based method demonstrates superior performance, providing new evidence of the potential of exploiting graph networks for the AVQA task.

**Comparison with large-scale pretraining-based methods.** The recent progress in the AVQA field tends to utilize either Large-Language Models (LLM) or large-scale data pretraining. **1)** AVLLM (Shu et al. 2023) and AVicuna (Tang et al. 2024) are two representative LLM-based methods that employ the Vicuna-13B and Vicuna-7B (Touvron et al. 2023), respectively. A common initial step in these methods is to align audio and visual features with the LLM’s token embedding space through linear projections. To achieve this, large-scale audio-caption and visual-caption pairs are used. Additionally, to adapt the LLM for processing audio and visual inputs and encoding cross-modal relations, these methods must fine-tune the LLM, which necessitates additional high-quality audio-visual-text data. As shown in Table 2, AVicuna requires 1.1M data in total. Since these

Method	Visual Encoder	Audio Encoder	A-QA	V-QA	AV-QA	Avg.
AVSD (Schwartz et al. 2019)	VGG-19	VGGish	68.78	70.31	65.44	67.32
PanoAVQA (Yun et al. 2021)	Faster R-CNN	VGGish	72.13	73.16	66.97	69.53
AVST (Li et al. 2022)	ResNet-18	VGGish	73.87	74.40	69.53	71.59
COCA (Lao et al. 2023)	ResNet-18	VGGish	75.42	75.23	69.96	72.33
TJSTG (Jiang and Yin 2023)	ResNet-18	VGGish	76.47	76.88	70.13	73.04
AV-Mamba (Huang et al. 2024)	ResNet-18	VGGish	74.67	79.27	70.62	73.63
SH-Mamba (Yang and Li 2024)	ResNet-18	VGGish	75.42	79.93	70.64	74.12
APL (Li et al. 2024)	DETR	VGGish	78.09	79.69	70.96	74.53
PSTP (Li, Hou, and Hu 2023)	CLIP <sub>B</sub>	VGGish	70.91	77.26	72.57	73.52
LAVISH (Lin et al. 2023)	Swin <sub>L</sub>	CLAP	75.40	79.60	70.10	73.60
CMP (Duan et al. 2024)	Swin <sub>L</sub>	CLAP	77.40	<b>81.90</b>	70.70	74.80
<b>PSOT<sub>B</sub> (ours)</b>	CLIP <sub>B</sub>	VGGish	<b>78.22</b>	80.07	<b>72.61</b>	<b>75.29</b>
<b>PSOT<sub>L</sub> (ours)</b>	CLIP <sub>L</sub>	CLAP	<b>79.08</b>	<b>87.12</b>	<b>74.07</b>	<b>78.42</b>

Table 1: **Comparison with mainstream methods.** The subscript ‘B’ in ‘PSOT<sub>B</sub>’ and ‘L’ in ‘PSOT<sub>L</sub>’ denote that the CLIP-ViT-B or CLIP-ViT-L model is used for visual feature extraction. The top-2 results are highlighted in **bold**.

Method	V- Enc.	A- Enc.	#PT	Params	Acc.
AVLLM (Shu et al. 2023)	CLIP <sub>L</sub>	CLAP	1.6M	13B	45.20
AVicuna (Tang et al. 2024)	CLIP <sub>L</sub>	CLAP	1.1M	7B	49.60
VALOR (Chen et al. 2023)	CLIP <sub>L</sub>	AST	33.5M	593M	78.90
<b>PSOT<sub>L</sub> (ours)</b>	CLIP <sub>L</sub>	CLAP	9k	8.3M	78.42

Table 2: **Comparison with large-scale pretraining-based methods.** ‘#PT’ denotes the number of samples used for model pretraining. ‘Params’ is the model size of LLMs (for the first two methods) or the trainable parameters of the entire model (for the last two methods).

Id	M-KPT	S-KPT	Q-KPT	A-QA	V-QA	AV-QA	Avg.
a	✓	✓	✓	79.08	87.12	74.07	78.42
b	–	✓	✓	78.09	87.12	73.01	77.65
c	✓	–	✓	77.53	86.99	73.50	77.79
d	✓	✓	–	77.84	86.21	73.38	77.57
e	–	–	✓	76.60	85.34	73.04	77.05
f	–	–	–	76.03	83.45	72.40	75.88

Table 3: **Ablation study on the proposed main modules.**

methods do not fine-tune on the downstream MUSIC-AVQA dataset, their zero-shot performances are currently far behind those of supervised learning-based methods shown in Table 1. 2) The recent method VALOR (Chen et al. 2023) focuses on audio-visual-language representation learning and utilizes 33.5M multimodal pairs for model pretraining. In this way, VALOR can generate advanced tri-modal features and generalize well to the AVQA task. After further fine-tuning on the MUSIC-AVQA dataset, VALOR achieves an impressive accuracy of 78.90%. In contrast, as shown in the Table, our method can achieve competitive performance while utilizing significantly less training data (9k vs. 33.5M) and fewer trainable parameters (8.3M vs. 593M). Notably, VALOR employs visual features ( $16 \times 16 \times 1024$ ) extracted by *fine-tuned* CLIP<sub>L</sub> model while we use the *downsampled* features ( $8 \times 8 \times 1024$ ) from *pretrained* CLIP<sub>L</sub>. These results validate the effectiveness of the proposed method, indicating that it is feasible to address specific downstream tasks by developing a resource-efficient expert model.

Id	$\mathcal{A}_t^m$	$\mathcal{A}_t^s$	A-QA	V-QA	AV-QA	Avg.
a	✓	✓	79.08	87.12	74.07	78.42
b	–	✓	79.14	86.75	73.52	77.92
c	✓	–	79.02	86.87	73.66	78.08
d	–	–	77.59	86.00	73.01	77.27

Table 4: **Impacts of the motion-/sound-driven adjacency matrices** in M-KPT and S-KPT modules, *i.e.*,  $\mathcal{A}_t^m$  and  $\mathcal{A}_t^s$ .

## Ablation Studies

**Ablation study on main modules.** We conduct ablation experiments to evaluate the effectiveness of the proposed main modules, *i.e.*, the M-KPT, S-KPT, and Q-KPT. The results are shown in Table 3. Compared to the complete model (a), the average QA performance after removing the M-KPT, S-KPT, or Q-KPT module (b/c/d) decreases by 0.77%, 0.63%, or 0.85%, respectively. When the M-KPT and S-KPT modules are simultaneously removed (e), the model performance further decreases, indicating that these modules performing in parallel complement each other. The model’s performance continues to deteriorate if all three modules are removed (f), *i.e.*, the audio and visual features extracted by backbones are directly sent to the final MMA module for answer prediction. These results clearly demonstrate the effectiveness of each proposed module.

**Impacts of  $\mathcal{A}_t^m$  and  $\mathcal{A}_t^s$  in M-KPT and S-KPT modules.** The  $\mathcal{A}_t^m$  (Eq. 3) and  $\mathcal{A}_t^s$  (Eq. 6) are the motion- and sound-driven adjacency matrices utilized in our M-KPT and S-KPT modules, respectively. To examine their impacts, we compare them with a vanilla strategy that directly calculates the feature similarity between original concatenated audiovisual nodes as the adjacency matrix, *i.e.*,  $\mathcal{A}_t = [v_t; a_t][v_t; a_t]^T$ . As shown in Table 4, compared to the full model (a), the average model performance without utilizing  $\mathcal{A}_t^m$  (b),  $\mathcal{A}_t^s$  (c), or both (d) all decreases. This proves that the motion intensity and audio-visual correspondence maps used for  $\mathcal{A}_t^m$  and  $\mathcal{A}_t^s$  computation are more useful for guiding graph learning.

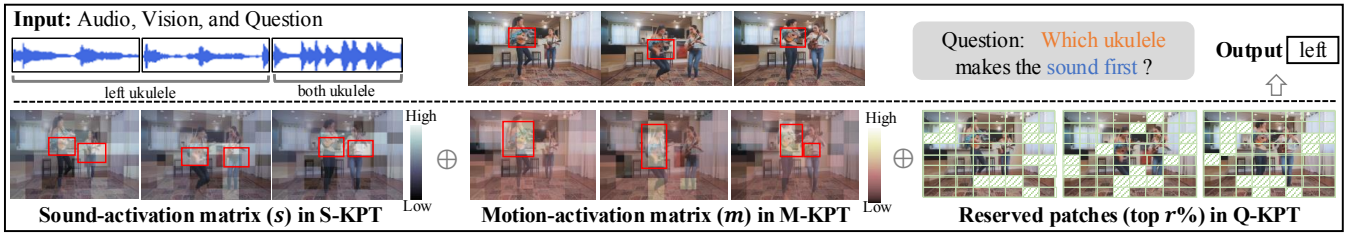


Figure 3: **Visualization of our model’s inference process for AVQA.** The key patches (red boxes) are highlighted in each module, guided by sound, motion, and question. The brighter color indicates larger activation weights.

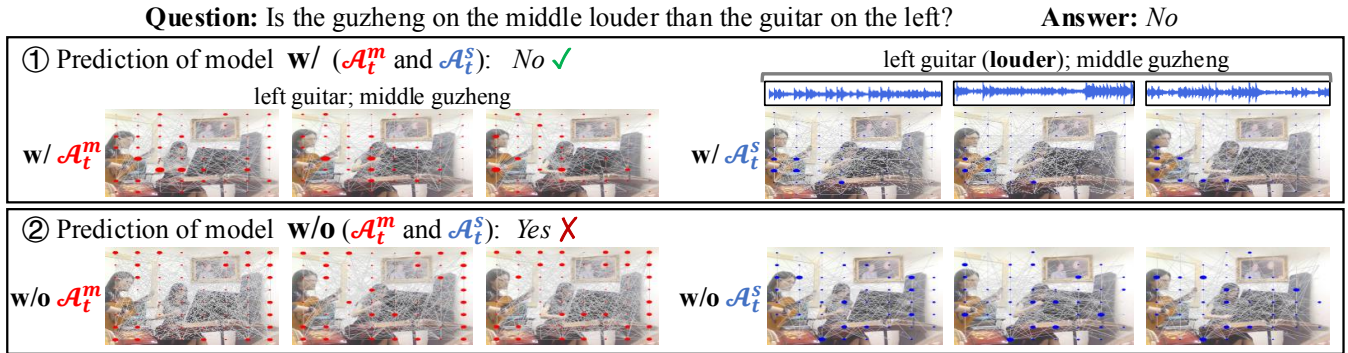


Figure 4: **Ablation visualization on the roles of  $A_t^m$  and  $A_t^s$  in our model.** The size of dots (●) located in the center of visual patches represents the aggregated weight from the adjacency matrix during the graph network learning.

## Qualitative Results

**Visualization of our model’s inference process.** We provide a qualitative example to illustrate how our model works for audio-visual question answering. As shown in Fig. 3, the person on the left continuously plays the ukulele, while the person on the right holds a ukulele for a time before playing it in the final segment. To answer the question “Which ukulele makes the sound first?”, our S-KPT module tracks the two instruments corresponding to the sound. As verified by the visualized sound-activation matrix  $s$  (Eq. 5), the visual patches of the two ukuleles exhibit larger weights (brighter colors). Since both ukuleles semantically correspond to the sound, it is challenging for S-KPT to identify the left ukulele as the source of the sound first. However, our M-KPT module aids in recognizing that the left ukulele is active in the first two segments due to its large movements, as evidenced by the motion-activation matrix  $m$  (Eq. 2). Afterward, the patches activated by the S-KPT and M-KPT are processed using the Q-KPT to retain only those patches highly relevant to the question. Some irrelevant visual patches, such as the floor and the leg of the person on the left, are removed at this stage. Through the collaboration of three modules, our model can focus on the most informative clues, ensuring accurate answers.

**Visualization on the impacts of the  $A_t^m$  and  $A_t^s$ .** Quantitative results in Table 4 demonstrate that our model, trained with motion- and sound-driven adjacency matrices, outperforms the model directly adopting the vanilla feature similarity as the adjacency matrix. In Fig. 4, we visualize the

learned adjacency matrices of both models. For each visual patch (node in the graph), we sum weights from all adjacent patches via  $A_t^m/A_t^s$ . A larger “●” indicates that the visual patch receives more attention (larger weights) in graph learning. The model trained with motion guidance (w/  $A_t^m$ ) accurately highlights and tracks the key patches related to the two sounding instruments; otherwise (w/o  $A_t^m$ ), it emphasizes many irrelevant patches, such as the window and wall. Similarly, the model trained with audio guidance (w/  $A_t^s$ ) pays more attention to the left guitar emitting louder sound; otherwise (w/o  $A_t^s$ ), the model focuses on the middle guzheng, resulting in an incorrect prediction.

## Conclusion

We propose a Patch-level Sounding Object Tracking (PSOT) method for the challenging AVQA task. By exploring patch-wise motion information across adjacent frames, we design a motion-driven Key Patch Tracking (KPT) module to track salient visual patches. Meanwhile, we propose a sound-driven KPT module to track sounding patches by exploring audio-visual correspondence. After identifying salient and sounding patches, we further utilize a question-driven KPT module to retain tracked patches highly relevant to the question. All these modules are implemented using graph neural networks, with audio-visual-question features automatically updated in graph learning. Extensive quantitative and qualitative results validate the effectiveness of our method.

## Acknowledgements

We sincerely thank the anonymous reviewers for their invaluable comments and insightful suggestions. We also extend our gratitude to Dr. Hui Wang for his helpful discussions. This work was supported by the National Natural Science Foundation of China (62272144, 72188101, and 62020106007), the Major Project of Anhui Province (2408085J040, 202203a05020011), and the Fundamental Research Funds for the Central Universities (JZ2024HGTG0309, JZ2024AHST0337, and JZ2023YQTD0072).

## References

- Biten, A. F.; Litman, R.; Xie, Y.; Appalaraju, S.; and Manmatha, R. 2022. Latr: Layout-aware transformer for scene-text vqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16548–16558.
- Chen, S.; He, X.; Guo, L.; Zhu, X.; Wang, W.; Tang, J.; and Liu, J. 2023. Valor: Vision-audio-language omni-perception pretraining model and dataset. *arXiv preprint arXiv:2304.08345*.
- De Oliveira Souza, B. C.; Aasan, M.; Pedrini, H.; and Rivera, A. R. 2023. SelfGraphVQA: a self-supervised graph neural network for scene-based question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4640–4645.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Duan, H.; Xia, Y.; Mingze, Z.; Tang, L.; Zhu, J.; and Zhao, Z. 2024. Cross-modal Prompts: Adapting Large Pre-trained Models for Audio-Visual Downstream Tasks. *Advances in Neural Information Processing Systems (NeurIPS)*, 1–10.
- Gu, A.; and Dao, T. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.
- Guo, D.; Li, K.; Hu, B.; Zhang, Y.; and Wang, M. 2024. Benchmarking Micro-action Recognition: Dataset, Methods, and Applications. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 6238–6252.
- Guo, R.; Ying, X.; Chen, Y.; Niu, D.; Li, G.; Qu, L.; Qi, Y.; Zhou, J.; Xing, B.; Yue, W.; Shi, J.; Wang, Q.; Zhang, P.; and Liang, B. 2023. Audio-Visual Instance Segmentation. *arXiv preprint arXiv:2310.18709*.
- Horn, B. K.; and Schunck, B. G. 1981. Determining optical flow. *Artificial Intelligence (AI)*, 185–203.
- Hu, R.; Singh, A.; Darrell, T.; and Rohrbach, M. 2020. Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9992–10002.
- Huang, Z.; Li, J.; Zhao, W.; Guo, Y.; and Tian, Y. 2024. AV-Mamba: Cross-Modality Selective State Space Models for Audio-Visual Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, 1–4.
- Jiang, Y.; and Yin, J. 2023. Target-Aware Spatio-Temporal Reasoning via Answering Questions in Dynamics Audio-Visual Scenarios. *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 9399–9409.
- Lao, M.; Pu, N.; Liu, Y.; He, K.; Bakker, E. M.; and Lew, M. S. 2023. COCA: COLlaborative CAusal Regularization for Audio-Visual Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 12995–13003.
- Li, G.; Hou, W.; and Hu, D. 2023. Progressive Spatio-temporal Perception for Audio-Visual Question Answering. In *Proceedings of the ACM international conference on multimedia (ACM MM)*, 7808–7816.
- Li, G.; Wei, Y.; Tian, Y.; Xu, C.; Wen, J.-R.; and Hu, D. 2022. Learning to Answer Questions in Dynamic Audio-Visual Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 19108–19118.
- Li, Z.; Guo, D.; Zhou, J.; Zhang, J.; and Wang, M. 2024. Object-Aware Adaptive-Positivity Learning for Audio-Visual Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 3306–3314.
- Lin, Y.-B.; Sung, Y.-L.; Lei, J.; Bansal, M.; and Bertasius, G. 2023. Vision transformers are parameter-efficient audio-visual learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2299–2309.
- Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 10012–10022.
- Mao, Y.; Shen, X.; Zhang, J.; Qin, Z.; Zhou, J.; Xiang, M.; Zhong, Y.; and Dai, Y. 2024. TAVGBench: Benchmarking text to audible-video generation. In *Proceedings of the ACM international conference on multimedia (ACM MM)*, 6607–6616.
- Park, J.; Lee, J.; and Sohn, K. 2021. Bridge to answer: Structure-aware graph interaction network for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15526–15535.
- Park, K. R.; Oh, Y.; and Kim, J. U. 2024. Enhancing Audio-Visual Question Answering with Missing Modality via Trans-Modal Associative Learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5755–5759.
- Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 8748–8763. PMLR.

- Schwartz, I.; Schwing, A. G.; Hazan, T.; et al. 2019. A simple baseline for audio-visual scene-aware dialog. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12548–12558.
- Shen, X.; Li, D.; Zhou, J.; Qin, Z.; He, B.; Han, X.; Li, A.; Dai, Y.; Kong, L.; Wang, M.; and Zhong, Y. 2023. Fine-grained audible video description. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10585–10596.
- Shu, F.; Zhang, L.; Jiang, H.; and Xie, C. 2023. Audio-Visual LLM for Video Understanding. *arXiv preprint arXiv:2312.06720*.
- Tang, Y.; Shimada, D.; Bi, J.; and Xu, C. 2024. AVicuna: Audio-Visual LLM with Interleaver and Context-Boundary Alignment for Temporal Referential Dialogue. *arXiv preprint arXiv:2403.16276*.
- Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 5998–6008.
- Wang, F.; Guo, D.; Li, K.; Zhong, Z.; and Wang, M. 2024. Frequency decoupling for motion magnification via multi-level isomorphic architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 18984–18994.
- Wu, Y.; Chen, K.; Zhang, T.; Hui, Y.; Berg-Kirkpatrick, T.; and Dubnov, S. 2023. 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)*, 1–5.
- Xiao, J.; Yao, A.; Liu, Z.; Li, Y.; Ji, W.; and Chua, T.-S. 2022. Video as conditional graph hierarchy for multi-granular question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2804–2812.
- Xiao, J.; Zhou, P.; Yao, A.; Li, Y.; Hong, R.; Yan, S.; and Chua, T.-S. 2023. Contrastive video question answering via video graph transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 13265–13280.
- Yang, Z.; and Li, e. a., Wenrui. 2024. SHMamba: Structured Hyperbolic State Space Model for Audio-Visual Question Answering. *arXiv preprint arXiv:2406.09833*.
- Yun, H.; Yu, Y.; Yang, W.; Lee, K.; and Kim, G. 2021. Pano-AVQA: Grounded Audio-Visual Question Answering on 360° Videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011–2021.
- Zhou, J.; Guo, D.; Guo, R.; Mao, Y.; Hu, J.; Zhong, Y.; Chang, X.; and Wang, M. 2024a. Towards Open-Vocabulary Audio-Visual Event Localization. *arXiv preprint arXiv:2411.11278*.
- Zhou, J.; Guo, D.; Mao, Y.; Zhong, Y.; Chang, X.; and Wang, M. 2024b. Label-anticipated Event Disentanglement for Audio-Visual Video Parsing. In *European Conference on Computer Vision (ECCV)*, 1–22.
- Zhou, J.; Guo, D.; and Wang, M. 2023. Contrastive positive sample propagation along the audio-visual event line. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 7239–7257.
- Zhou, J.; Guo, D.; Zhong, Y.; and Wang, M. 2023. Improving audio-visual video parsing with pseudo visual labels. *arXiv preprint arXiv:2303.02344*.
- Zhou, J.; Guo, D.; Zhong, Y.; and Wang, M. 2024c. Advancing Weakly-Supervised Audio-Visual Video Parsing via Segment-wise Pseudo Labeling. *International Journal of Computer Vision (IJCV)*, 1–22.
- Zhou, J.; Shen, X.; Wang, J.; Zhang, J.; Sun, W.; Zhang, J.; Birchfield, S.; Guo, D.; Kong, L.; Wang, M.; and Zhong, Y. 2024d. Audio-visual segmentation with semantics. *International Journal of Computer Vision (IJCV)*, 1–21.
- Zhou, J.; Wang, J.; Zhang, J.; Sun, W.; Zhang, J.; Birchfield, S.; Guo, D.; Kong, L.; Wang, M.; and Zhong, Y. 2022. Audio-visual segmentation. In *European Conference on Computer Vision (ECCV)*, 386–403.
- Zhou, J.; Zheng, L.; Zhong, Y.; Hao, S.; and Wang, M. 2021. Positive sample propagation along the audio-visual event line. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8436–8444.