GHR-VQA: Graph-guided Hierarchical Relational Reasoning for Video Question Answering

Dionysia Danai Brilli [1,2], Dimitrios Mallis [3], Vassilis Pitsikalis [4] and Petros Maragos [1,2]
(1) School of ECE, National Technical University of Athens, Athens, Greece
(2) Robotics Institute, Athena Research Center, Athens, Greece
(3) University of Luxembourg, Kirchberg, Luxembourg
(4) deeplab.ai, Athens, Greece

***Abstract*—We propose GHR-VQA, Graph-guided Hierarchical Relational Reasoning for Video Question Answering (Video QA), a novel framework that incorporates scene graphs to capture intricate object relationships and interactions within video sequences. Unlike traditional pixel-based methods, our approach processes scene graphs with Graph Neural Networks (GNNs), transforming structured video representations into rich, context-aware embeddings for efficient processing. By leveraging scene graphs, our model inherently enhances interpretability and enables a more profound understanding of spatiotemporal dynamics, addressing limitations of existing Video QA models that fail to fully interpret object interactions. Our model employs a hierarchical network to reason across different abstraction levels, enhancing both local and global understanding of video content. We validate our approach on the Action Genome Question Answering (AGQA) dataset, achieving significant performance improvements in some question categories. Notably, our method excels in object-relation reasoning, surpassing SOTA by 9.8%.**

***Index Terms*—Video Question Answering, Scene Graphs, Graph Neural Networks, Hierarchical Conditional Relation Network, Action Genome Question Answering**

I. Introduction

In the rapidly evolving digital era, the exponential growth in video content has accentuated the need for sophisticated tools capable of interpreting complex video data for various applications. Video Question Answering (VideoQA) emerges as a critical domain, providing a powerful framework for enabling machines to reason about multimedia content and unlocking a wide array of practical applications.

Despite continuous advancements, the field still faces significant challenges. A key difficulty lies in effectively capturing the intricate interconnections between entities in a scene and their evolution over time. The temporal dimension increases complexity, with existing approaches often struggling to capture and interpret dynamic object relationships within videos [18]. A line of research [9] explores learning of situation hyper-graphs for videos by extracting latent graph representations from raw visual scenes. These intermediate representations enable effective modeling of entities and their relationships, facilitating a deeper understanding of the video's semantic content and enhancing question-answering performance. Still, such modeling pipelines are highly resource-intensive as they involve complex architectures and large-scale models. Thus, training requires extensive annotated VideoQA datasets which may not always be readily available.

This work introduces a novel modeling approach for VideoQA that departs from learning latent situation graph representations from video inputs. Instead, we leverage an off-the-shelf Scene Graph Generation (SGG) model to directly infer scene graphs from video frames. Using these explicit representations of object interconnections within the scene, we propose a lightweight framework for VideoQA that relies on encoding frame-level scene graphs. Our approach eliminates dependence on raw video frames, enabling question generation solely based on the extracted scene graphs. As highly compact representations, scene graphs facilitate lightweight yet effective modeling of temporal dynamics, enabling robust performance with a minimal VideoQA model.

Our pipeline, GHR-VQA, begins with an efficient Scene Graph Generation (SGG) step using the state-of-the-art method from [23]. The generated scene graphs serve as input to the Scene Graph Encoding Module (SGEM), which maps them to latent representations through a shallow Graph Neural Network (GNN). These graph representations, combined with BERT-based question encodings, are then processed by a hierarchical network to produce conditioned embeddings and the final answer. We evaluated our method in the Action Genome Question Answering Dataset [6] where we demonstrate robust performance via a simple and lightweight modeling approach.

II. Related Works

A. Video Question Answering

Video Question Answering is an emerging task at the intersection of computer vision and natural language processing. It requires models to reason about spatiotemporal information in videos to answer questions correctly. Early approaches relied on recurrent neural networks (RNNs) and convolutional neural networks (CNNs) to process temporal and spatial features, respectively [14], [22]. More recent advances have incorporated attention mechanisms and transformers, enabling long-range dependency modeling and improved multimodal reasoning [4], [24]. Many contemporary techniques approach this task with Large Language Models (LLMs) and Large Visual-Language Models (LVLMs) [10], [16]. These pre-trained models learn from a massive amount of diverse data and can handle unseen data effectively. Despite these advancements, existing methods often struggle with capturing detailed object interactions and

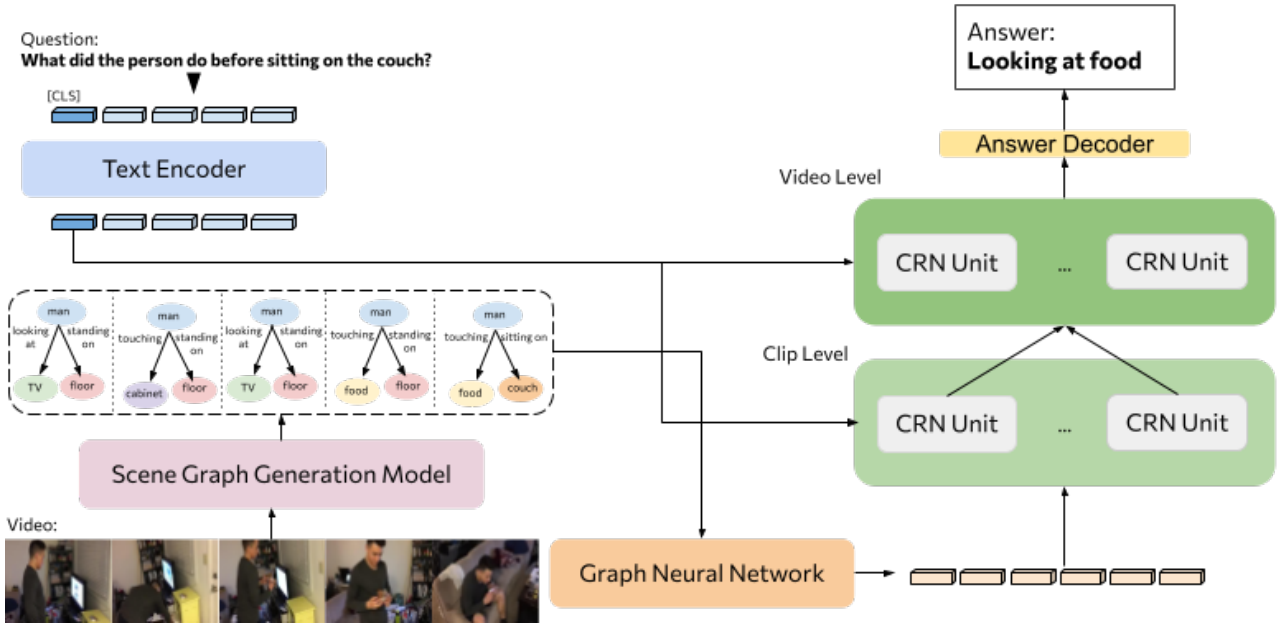

Fig. 1. Our proposed architecture. The process begins with the input of a question and a corresponding video. Initially, we perform clip selection and pass the segments through an SGG model to extract scene graphs that represent the visual elements and their interrelationships. These extracted scene graphs are processed by a GNN, which generates meaningful embeddings. The embeddings are then fed into a hierarchical network, which integrates and contextualizes the information across different levels of abstraction to generate a comprehensive understanding in relation to the query and finally answer the question.

higher-level reasoning, and often rely upon language biases, motivating research into structured representations[8].

B. Graphs in Video QA

A promising direction of Video Question Answering is using graph-based methods, where relationships between objects, actions and attributes are explicitly modeled. Graph-based models offer advantages in interpretability and relational reasoning but can be computationally expensive and reliant on accurate object detection. Many methods leverage scene-graphs, which represent objects, relationships and attributes as structured triplets, since they capture semantic relationships explicitly [2], [7], [21]. Khan et al. introduced situation hypergraphs, training a decoder to implicitly identify graph representations with actions and human-object relationships [9]. There have also been attempts with 3D scene graphs, capturing the objects within a dynamic spatiotemporal graph in a 3D space [1]. Although significant progress has been made in VideoQA through graph-based reasoning, learning structured semantic situational representations from videos remains a challenging problem. Our approach departs from these works as it is a lightweight solution to VideoQA that operates on explicit scene graphs. It provides a transparent reasoning process that can be visualized and analyzed, reducing computational overhead thanks to processing graphs compared to raw video frames. Using a hierarchical model we capture video content in two granularities while minimizing reliance on language.

III. Methodology

A. Problem Formulation

This work follows a classification formulation for the VideoQA task as in [6]. The objective is to predict the correct answer $a_i^* \in A$ over a fixed set of K possible answers or $A = \{a_1, a_2, ..., a_k\}$, for an input video $V_i \in \mathcal{V}$ representing a sequence of N frames $V_i = [f_1, f_2, ..., f_N]$ and a question $q_i \in Q$. The goal becomes to learn a mapping function $\mathcal{F} : \mathcal{V} \times Q \rightarrow A$, that predicts the correct answer or $a_i^* = \mathcal{F}(V_i, q_i)$.

B. Proposed Approach

The mapping $\mathcal{F}$ is formed through a series of steps.

Frame Scene Graph Generation: In the core of GHR-VQA is an initial frame scene graph generation step for each individual frame of the input video. A scene graph $\mathcal{G}_i = (\mathcal{N}_i, \mathcal{E}_i)$ serves as a structured representation of a scene captured within the frame, where nodes $\mathcal{N}_i$ correspond to the entities or objects, and the edges $\mathcal{E}_i$ represent semantic relationships between these entities. Each node $n_j^i \in \mathcal{N}_i$ includes the bounding box coordinates of the entity and and the corresponding class label, while edges $e_{j,k}^i \in \mathcal{E}_i$ capture the type of semantic relationship between two objects n_j^i and n_k^i. For example, given a frame showing a person holding a cup, the scene graph would consist of two nodes representing the ”person” and the ”cup” and an undirected edge labeled ”holding” between them.

We start by processing each input frame f_i individually via a Scene Graph Generation Module or $\mathcal{G}_i = \mathcal{S}_g(f_i)$. For $\mathcal{S}_g$ we use the off-the-shelve scene graph generator of [19], consisting of a pre-trained detector backbone and the scene

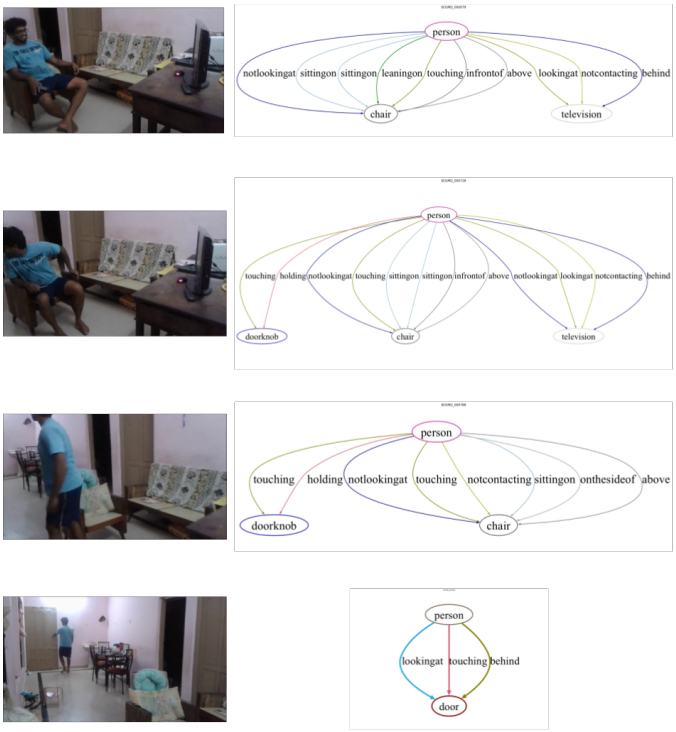

Fig. 2. Example of 4 frames from the video sample OCGMQ with the corresponding annotated scene graphs.

generation module from MOTIFS [23]. $\mathcal{S}_g$ is trained on the Visual Genome dataset [11] containing 150 object categories and 50 types of relationships / predicates.

Frame Scene Graph Encoding: Generated scene graphs for video frames are then encoded into a latent scene representation via a Scene Graph Encoding Module (SGEM) modeled as a 2-layer Graph Neural Netwrok (GNN). Specifically video frames are first aggregated into a hypergraph $G = \{\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_N\}$ for the N video frames which is processed by a Heterogeneous Edge Graph Attention Network (HetEdgeGAT) to learn cross-entity semantic patterns. Our SGEM follows [15] where each node n_j^i in graph $\mathcal{G}_i$ is updated as:

$$(n_j^i)' = \text{EdgeGAT}_r(n_j^i) = \Theta_{s,r} \cdot n_j^i + \Big\|_{h=1}^{H} \left(\sum_{k \in \mathcal{N}_r(n_j^i)} \alpha_{j,r,k}^h \left(\Theta_{v,r}^h \cdot n_k^i + \Theta_{e,r}^h \cdot e_{j,r,k} \right) \right) \tag{1}$$

where Θ is used to denote learnable weight matrices for the transformation of features of the node to update self (s), neighboring nodes (u) and edge features (e) for graph $\mathcal{G}_i$. H corresponds to the number of attention heads and the $\|$ denotes the concatenation operator. Attention weights are obtained by:

$$\alpha_{j,r,k}^h = \text{softmax}_{r,k}\left(\text{LeakyReLU}\left(a_r^{h^T} \left[\Theta_{v,r}^h \cdot n_j^i \parallel \Theta_{v,r}^h \cdot v_k^i \parallel \Theta_{e,r}^h \cdot e_{j,r,k} \right] \right)\right) \tag{2}$$

with a corresponding to a learnable vector.

We use cascaded layers in order to aggregate information in the frame regarding all types of relationships. To get a comprehensive representation of the entire graph representation, we combine the individual embeddings of the nodes into a single vector. This aggregation is achieved through summing the node embeddings for each node n_j^i in the graph $\mathcal{G}_i$ are aggregated using the following function: $\mathcal{G}_i = \mathcal{F}\left(\left\{n_j^i | j \in \mathcal{N}_i\right\}\right)$ where $\mathcal{G}_i$ is the final graph embedding and $\mathcal{F}$ is a sum function $\mathcal{F} = \sum_{j \in N} n_j^i$.

Question Encoder In addition to graph processing, we map the input question into a latent space by leveraging token-wise sentence embeddings extracted from the penultimate layer of a BERT model [3]. Specifically, we utilize the [CLS] token embedding from the model's output, for a holistic representation of the entire sentence.

Hierarchical Network We propose a hierarchical method to process graph embeddings at different levels of granularity to finally classify the answer to the question. Drawing inspiration from [12], we adapt the model to integrate scene graphs, thus benefiting from the hierarchical and contextual processing of the CRN units. The core of the Hierarchical Network consists of multiple Conditional Relational Network (CRN) units arranged hierarchically. The CRN units at the lower level process data at the clip level, gathering information from multiple frames and handling more spatial information, whereas the CRN units at the higher level operate at video level, modeling longer temporal dependencies. The hierarchical design enables the model to consider information in different contexts.

The top-level CRN layer outputs a video graph embedding, used to classify the answer. This video-graph embedding is aggregated with the question embeddings and is processed by an answer decoder that generates the final output.

Each CRN unit takes as an input an array of n objects $\mathcal{X} = (x_1, ..., x_n)$ and a conditioning feature c as a global context. The objects belong in the same vector space R^d. CRN computes a relation-based transformation y_i conditioned on feature c:

$$y_i = \mathcal{F}(x_i, c) \tag{3}$$

The input array $\mathcal{X}$ is first processed to model k-tuple relations from t sub-sampled size-k subsets by sub-network g^k. The outputs are conditioned with the context $\rfloor$ via sub-network h^k and finally aggregated by p^k to obtain a result vector r^k which represents k-tuple conditional relations.

IV. Experiments

A. Dataset

We evaluate our method on the Action Genome Question Answering Dataset 2.0 [6]. Action Genome Question Answering (AGQA) is a benchmark for compositional spatio-temporal

Method	obj-rel	rel-action	obj-action	superlative	sequencing	exists	duration	activity	Overall
Results on Dataset Subset A (our subset)									
PSAC [13]	37.84	49.95	50.00	33.20	49.78	49.94	45.21	4.14	40.18
HME [5]	37.42	49.90	49.97	33.21	49.77	49.96	47.03	5.43	39.89
HCRN [12]	40.33	49.86	49.85	33.55	49.70	50.01	43.84	5.52	42.11
GHR-VQA (ours)	**49.8**	53.7	55.1	**40.9**	50.4	**56.3**	25.5	7.4	42.5
Results on Dataset Subset B (full AGQA dataset)									
SHG-VQA [9]	46.42	**60.67**	**64.63**	38.83	**62.17**	56.06	**48.15**	**10.12**	**49.20**

TABLE I
COMPARISON OF SOTA METHODS ON DIFFERENT AGQA DATASET SUBSETS. DATASET SUBSET A CORRESPONDS TO OUR EXPERIMENTAL SUBSET, WHILE DATASET SUBSET B REFERS TO THE COMPLETE AGQA DATASET.

reasoning. This benchmark contains 96.85M question-answer pairs and a balanced subset of 2.27M question-answer pairs.

The dataset comprises approximately 9.6k videos, each with a duration of 30 seconds, recorded at a frame rate of 30 frames per second (fps). This translates to an average of around 900 frames per video. However, a notable aspect of AGQA is the selective annotation process applied to these videos. The dataset's goal, extending Action Genome [8], is to decompose actions, so its focus is on annotating only the video segments where actions occur and objects are involved in it. Thus, 5 frames uniformly sampled across each action interval are annotated. For each annotated frame, there is information for the visible objects, their bounding boxes, their labels, and relationships between them. So, despite the large number of frames available per video, on average, only 35 frames per video are annotated. This approach underscores the dataset's emphasis on specific, salient moments within the videos, rather than an exhaustive frame-by-frame annotation with redundant objects and relationships.

To accommodate various computational capacities and enable faster while detailed analysis, we designed a distinct experimental framework. This framework was designed to maintain the original dataset's distribution through random sampling, ensuring an accurate representation of the AGQA dataset's diversity and complexity. Our training subset, mentioned in Table I as Dataset Subset A, corresponds to 100K train question-answer pairs and 20K test question-answer pairs. We ensured no data leakage from the train to the test set by keeping different videos in each set. To validate the effectiveness of our experimental setup, we benchmarked it against the baseline models provided by the AGQA creators. Our evaluation confirms that these models achieve consistent performance across both the full dataset and our subset, serving as a grounding for subsequent analysis.

B. Implementation Details

In our experimental setup, we utilized PyTorch[17] as the main framework for all model training and development. For graph-related tasks, especially in Graph Neural Networks (GNNs), we employed the Deep Graph Library (DGL) [20], which is compatible with PyTorch and provides optimized graph data structures and operations. All our experiments were run using two machines, each equipped with four GPUs.

V. RESULTS & ANALYSIS

As we can see in Table I, our approach places second in overall score among the state- of-the-art methods. Our approach presents comparable results in almost all question categories and even outperforms in some of them.

The best baseline on AGQA is HCRN for overall accuracy. HCRN uses appearance features from ResNet101 as well as motion features from ResNext101-Kinetics400 backbones. Our model outperforms HCRN by almost 0.5%. However, we observe the biggest improvement of 9.8% absolute points on object-relation reasoning questions compared to the best baseline in that category and 3.38% absolute points on the best-performing model, SHG-VQA.

First of all, our approach achieves the highest accuracy in 'obj-rel' category, meaning our model can efficiently understand relationships between objects within the scene. Our model also performs best in 'exists' and 'superlative' categories which means it can accurately identify the occurrence of concepts and objects and their order. In the rest of the categories -except for duration-, our model is ranked second. However, we can infer that the model struggles with temporal reasoning, particularly in the 'duration' category, highlighting challenges in capturing temporal dependencies.

Moving on to a qualitative assessment of our method, we can notice in Figure 2 that the generated scene graphs can capture the semantic information from each frame, and thus our GNN learns to extract meaningful embeddings so that the HCRN classifies the answer accordingly.

Model	Accuracy (%)
GINE + MLP	32.8
EdgeGAT + MLP	34.6
HetEdgeGAT + MLP	33.9
HetEdgeGAT + HCRN (GHR-VQA)	42.5

TABLE II
ACCURACY COMPARISON OF DIFFERENT GNN ARCHITECTURES ALONG WITH MLP OR HCRN.

Finally, the results presented in Table II provide a comparative analysis of the accuracy achieved by different GNN architectures integrated with either an MLP or the HCRN module. These experiments highlight the advantages of our proposed approach in leveraging heterogeneous edge-based attention mechanisms for Video QA. Among the evaluated architectures,

our final method, GHR-VQA, pairing the HetEdgeGAT model with HCRN, achieves the highest accuracy of 42.5%, outperforming other combinations. This significant improvement demonstrates the complementary strengths of HetEdgeGAT and HCRN; Our SGEM introduces heterogeneous edge-based attention, effectively modeling diverse relationships between objects and actions and enabling richer semantic understanding and the hierarchical network further enhances reasoning capabilities by capturing spatiotemporal dependencies across different levels. Furthermore, the incremental gains observed when comparing GNN architectures emphasize the importance of modeling edge heterogeneity and employing advanced attention strategies for improved graph-based reasoning. These findings validate the robustness and scalability of our proposed HetEdgeGAT + HCRN framework, showcasing its ability to handle the complexity of visual reasoning tasks.

VI. Discussion & Conclusion

Our scene-graph guided hierarchical model for Video QA achieves notable improvements in object-relation reasoning, outperforming existing methods in this category. However, the computational cost of scene graph generation and GNN processing can be prohibitive, especially for longer videos, and the accuracy of object detection heavily impacts the quality of the generated scene graphs.

Future work should focus on enhancing temporal reasoning by integrating more advanced models for better sequencing. Additionally, refining scene graph generation and improving scalability through techniques like graph sparsification could improve efficiency and performance.

In conclusion, our scene-graph-guided hierarchical model represents a step forward in the Video QA field. By moving beyond pixel-based approaches and incorporating relational reasoning through scene graphs, our framework achieves strong results on the AGQA dataset. While temporal reasoning remains a challenge, our work lays the foundation for more nuanced and scalable video understanding, with potential applications in healthcare, autonomous vehicles and beyond.

References

[1] A. Cherian, C. Hori, T. K. Marks, and J. Le Roux, "(2.5+1)d spatio-temporal scene graphs for video question answering," in *Proc. AAAI*, 2022.

[2] L. H. Dang, T. M. Le, V. Le, and T. Tran, "Hierarchical object-oriented spatio-temporal reasoning for video question answering," *Proc. IJCAI*, 2021.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL*, Association for Computational Linguistics, 2019.

[4] C. Fan, X. Zhang, S. Zhang, W. Wang, C. Zhang, and H. Huang, "Heterogeneous memory enhanced multimodal attention model for video question answering," in *Proc. CVPR)*, 2019.

[5] C. Fan, X. Zhang, S. Zhang, W. Wang, C. Zhang, and H. Huang, "Heterogeneous memory enhanced multimodal attention model for video question answering," in *Proc, CVPR*, 2019.

[6] M. Grunde-McLaughlin, R. Krishna, and M. Agrawala, "AGQA: A benchmark for compositional spatio-temporal reasoning," in *Proc. CVPR*, 2021.

[7] R. Hu, A. Rohrbach, T. Darrell, and K. Saenko, "Language-conditioned graph networks for relational reasoning," in *Proc. ICCV*, 2019.

[8] J. Ji, R. Krishna, L. Fei-Fei, and J. C. Niebles, "Action genome: Actions as compositions of spatio-temporal scene graphs," in *Proc. CVPR*, 2020.

[9] A. U. Khan, H. Kuehne, B. Wu, *et al.*, "Learning situation hyper-graphs for video question answering," in *Proc. CVPR*, IEEE, 2023.

[10] W. Kim, C. Choi, W. Lee, and W. Rhee, "An image grid can be worth a video: Zero-shot video question answering using a vlm," *IEEE Access*, 2024.

[11] R. Krishna, Y. Zhu, O. Groth, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, vol. 123, pp. 32–73, 2017.

[12] T. M. Le, V. Le, S. Venkatesh, and T. Tran, "Hierarchical conditional relation networks for video question answering," 2020.

[13] X. Li, J. Song, L. Gao, *et al.*, "Beyond RNNs: Positional self-attention with co-attention for video question answering," *Proc. AAAI*, 2019.

[14] L. Ma, Z. Lu, and H. Li, "Learning to answer questions from image using convolutional neural network," *Proc. AAAI*, 2016.

[15] T. Monninger, J. Schmidt, J. Rupprecht, *et al.*, "Scene: Reasoning about traffic scenes using heterogeneous graph neural networks," *IEEE Robotics and Automation Letters*, 2023.

[16] J. Pan, Z. Lin, Y. Ge, *et al.*, "Retrieving-to-answer: Zero-shot video question answering with frozen large language models," in *Proc. ICCV Workshop*, Oct. 2023, pp. 272–283.

[17] A. Paszke, S. Gross, F. Massa, *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, 2019.

[18] S. Paul, K. Rao, G. Coviello, *et al.*, "Why is the video analytics accuracy fluctuating, and what can we do about it?" In *Proc. ECCV Workshops*. Springer Nature Switzerland, 2022.

[19] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *Proc. CVPR*, 2020.

[20] M. Wang, L. Yu, D. Zheng, *et al.*, "Deep graph library: Towards efficient and scalable deep learning on graphs," *Proc. ICLR*, 2019.

[21] J. Xiao, A. Yao, Z. Liu, Y. Li, W. Ji, and T.-S. Chua, "Video as conditional graph hierarchy for multi-granular question answering," in *Proc. AAAI*, 2022.

[22] C. Yin, J. Tang, Z. Xu, and Y. Wang, "Memory augmented deep recurrent neural network for video question answering," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[23] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proc. CVPR*, 2018.

[24] Z. Zhao, Q. Yang, D. Cai, *et al.*, "Video question answering via hierarchical spatio-temporal attention networks," in *IJCAI*, 2017.