Multi-scale Joint Attention Network for Video Question Answering

Anonymous Author(s)
Submission Id: 1973*

ABSTRACT

Video question answering (VideoQA) is a challenging multi-modal task, which aims to find the correct answer given a video and a question in natural language form. The key problem of VideoQA is to find the local visual information related to the question in spatial and temporal dimensions. Despite being the most popular, conventional methods typically explore and pay more attention to the global features. As a result, they fail to comprehensively and accurately describe the visual objects when the scale is vast and their spatial-temporal position variations. In this paper, we propose a Multi-scale Joint Attention Network (MJAN) to solve the VideoQA problem in a unified framework. The object detection-based feature extractor is proposed to obtain the multi-scale visual features. Then, a new attention module is designed to analyze the joint correlation between the visual and textual features in the spatio-temporal dimensions coordinately. To evaluate the performance of our model, we conduct experiments on the TGIF-QA, MSVD-QA, and MSRVTT-QA datasets. Extensive experimental results show that the proposed MJAN outperforms state-of-the-art methods by a large margin.

CCS CONCEPTS

• **Computing methodologies** → Visual content-based indexing and retrieval.

KEYWORDS

deep learning, multi-modal, video question answering, attention mechanism

1 INTRODUCTION

The task of combining computer vision with natural language has received extensive attention. Visual question answering (VQA) [2, 11, 17, 21] is a typical visual-textual integration example, which aims to generate the correct answer given an image/video and a question in natural language. According to the different visual sources, visual question answering can be divided into image question answering (ImageQA) and video question answering (VideoQA).

Among the these two tasks, ImageQA has been extensively studied from various strategies, including visual information understanding, textual information understanding, and multi-modal information fusion [1, 2, 35]. ImageQA has been the primary basis for VQA, where the same strategies are used to provide spatial information descriptions. The data of the spatial dimension are critical for VQA task. It is noted that the details of images should be comprehensively taken into account for a more effective visual information extraction.

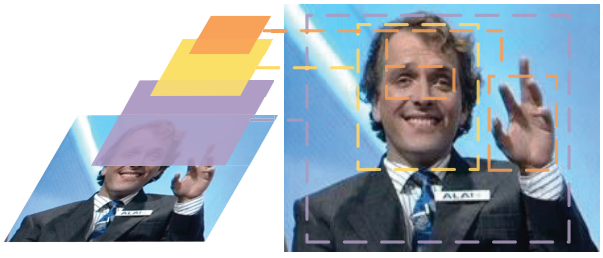

Figure 1: For the VideoQA task, different questions can be raised for one video, and the extracted visual features should describe all kinds of information as comprehensively as possible. The proposed Multi-scale Joint Attention Network (MJAN) in this paper employs a multi-scale feature extractor with a joint attention mechanism in the spatio-temporal dimensions to adapt to various questions.

As the visual source is video, both spatial and temporal dimensions are involved in VideoQA, making it more challenging than ImageQA. To answer various questions as shown in Fig. 1, motion information including the class and context of actions should be considered, in addition to appearance information such as the position, the color, or the number of objects. To this end, there are many advancements in VideoQA models. Jang *et al.* [17] employed ResNet and C3D for statistic information and dynamic information description, respectively. Gao *et al.* [7] introduced optical flow as extra model input to represent motion information. Gao *et al.* [8] obtained the temporal information from feature map sequences by LSTM structure. Generally, extracting the visual features in video comprehensively and effectively is the key to generate a correct answer.

However, most existing works only pay attention to the global information [5, 7, 8, 17, 20], while local features which are more informative have been unfortunately neglected. The feature extractor is pre-trained on the classification task, which only focuses on a single object in the input image. As shown in Fig. 1, the features of 'man' are easy to be obtained, while the 'hand' and 'eye' are likely

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM MULTIMEDIA '21, October 20–24, 2021, Chengdu, CN
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

to be ignored. Some research [16, 33] has attempted to address the absence of local information by using the high-confidence objects detected by a network. The detection results which are the location and the category of the object are adopted. However, these methods are only effective when the objects are well described during a pre-training process. The fine-grained degree of annotation will greatly affect the results. For example, the detection network can not take the 'hand' as an object if 'hand' is not labeled in the pre-training dataset. To address this issue, we leverage the multi-scaled feature extracted by the object detection network to represent the input video. To the best of our knowledge, it is the first attempt to explore pyramid-level features for the VideoQA task. Instead of the global level or object level, we represent the video by feature map level, as shown in Fig. 1. The rectangles with different color are the symbolistic representations of the multi-scale feature maps. The proposed method avoids mentioned deficiency of the model caused by the pre-training.

The multi-scale feature maps provide global and local information in the spatial dimension. Besides, the temporal dimension in the video should be handled delicately. Thus, the spatio-temporal clue localization for a question is another key issue for generating a correct answer. Recent methods explored the application of attention mechanisms for VideoQA. Gao *et al.* proposed a co-memory attention structure for appearance and motion sequences [7]. Li *et al.* introduced a self-attention from NLP into VideoQA [23]. Spatial information has not been well utilized in most recent methods. In Jang *et al.*, the spatio-temporal attention model performs even worse than the temporal attention only model [17]. To address this issue, we adopt the hypothesis that the fusion of visual and textual features should consider jointly in both temporal and spatial aspects. We address the issue by not only analyzing 'when to look' but also 'where to look.' Thus, we lead a Multi-scale Joint Attention Network (MJAN), consisting of a Multi-scale Feature Extraction Module (MFEM) and a Joint Attention Module (JAM) for the VideoQA task. The main contributions are summarized as follows:

- We introduce a Multi-Scale Feature Extraction Module (MFEM) to comprehensively employ the spatial visual information of different scales to describe the input video. To the best of our knowledge, it is the first attempt to take pyramid-level features as the backbone for the VideoQA task.
- We introduce a Joint Attention Module (JAM), which is based on the attention mechanism to fully exploit both temporal and spatial features to answer the question.
- The proposed method is extensively evaluated on TGIF-QA, MSVD-QA, and MSRVTT-QA datasets, showing significantly superior performance in comparison with state-of-the-art models.

2 RELATED WORK

In this section, we briefly review the representative works for the visual question answering.

2.1 Image Question Answering

The ImageQA task requires the network to infer the answer according to the given image and question. It provides the basic idea about the feature representation and the fusion of the video and the text. Early work in [2] introduced a simple baseline for the ImageQA task, which was a typical CNN-RNN structure network. It took CNN based network to represent visual information and RNN based network to obtain the textual feature. The work in [29] proposed the VIS-LSTM model, which combined the image sequences and text by LSTM structure. Yang *et al.* [35] designed the Stacked Attention Network (SAN), which can filter the image region related to the answer. In [9], Gao *et al.* introduced the bilinear pooling method, which can fuse the multi-modal features by the outer product operation. Furthermore, the work in [6] employed both compact bilinear pooling and spatial attentions. In [1, 31], Anderson *et al.* proposed the bottom-up and top-down attention for object features detected by Faster-RCNN. In [27], the difference of the attention and the non-attention parts were connected. The attention and the non-attention parts were separated by an appropriate distance margin in a feature embedding space. The correlation between the attention and the non-attention parts were enforced as a constraint for attention learning. It fails to specify a certain object that is related to the question. The methods for ImageQA inspire the research of VideoQA.

2.2 Video Question Answering

The VideoQA task is more complex and challenging due to the extra information in the temporal dimension. Based on the methods for ImageQA, the work for VideoQA mainly focused on extracting temporal features and the improvements of attention mechanism. The work in [17] published the first large-scale VideoQA dataset TGIF-QA and introduced a basic VideoQA model Spatial-Temporal Network (ST). Xu *et al.* [33] published MSVD-QA and MSRVTT-QA datasets and designed gradually refines attention (GRA) for appearance and motion features. Gao *et al.* [7] took optical flow features into account and proposed the co-memory attention network, which can analyze the static information from ResNet and dynamic information from optical flow at the same time. The work in [23] employed a self-attention mechanism on both frame sequence (video) and word sequence (question). Gao *et al.* [8] divided the video into multiple segments, then the question-related features were obtained by Structured Two-stream Attention (STA) for each segment. Both Jiang *et al.* [19] and Huang *et al.* [16] introduced a graph convolutional network (GCN) for the VideoQA task. The GCN can analyze the visual information consisting of global features and object features more effectively.

In [30], Song *et al.* focused on the visual reasoning problem in the video question answering task. A new system is designed to automatically generate a new dataset for multi-step reasoning. The attention module was designed to address the sub-tasks embedded in questions. The long-term temporal dependency was captured by the GRU. The temporal and the spatial dimensions are modeled separately. We propose an attention module analyzing the joint correlation between the visual and textual features. In [22], the rich video and the question information were represented by aggregating the frame-level features adaptively. The bags-of-words quantization was adopted. The most related signals were extracted to predict the answer. In [37], the diverse features are extracted from the information containing appearance, motion, and audio

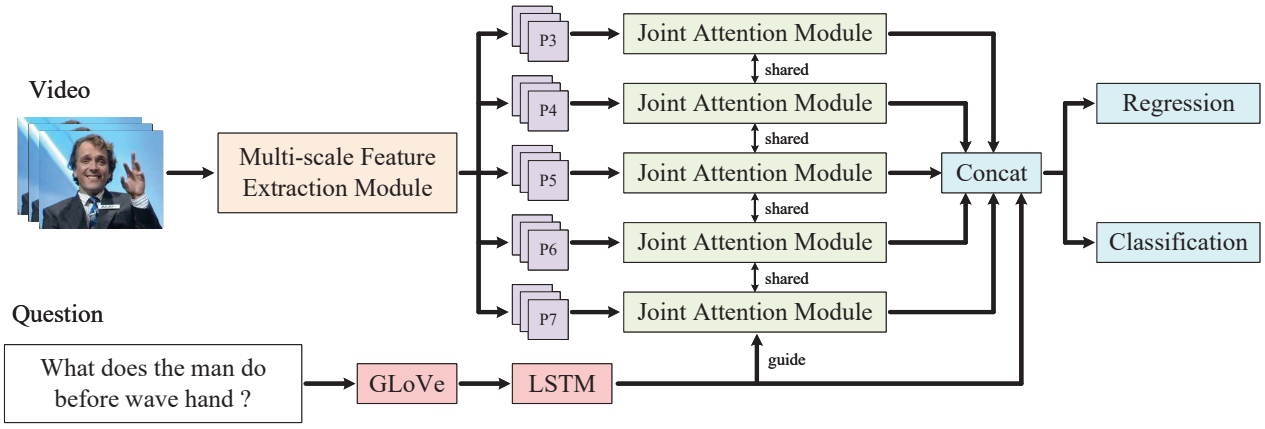

Figure 2: The proposed framework of Multi-scale Joint Attention Network (MJAN) for the VideoQA task.

data. Moreover, the relevant text information were obtained from Wikipedia. The multi-channel features were analyzed in the reasoned union to support the correct answer. The frame level feature is adopted, details in the object were not mined. Thus, our proposed method employs the local feature to improve the performance of spatial representation. In [4], the objects were detected firstly, the video was turned from the spatio-temporal unit into an expected relational graph representing the object. The information was adopted in the systematic reasoning network to produce the answer to the question. Unlike existing work, we use pyramid-level features to represent the visual information and propose a joint attention mechanism to fuse the visual and textual features. This method need to focus on objects in advance, which will probably lead to the failure of extracting those necessary features. To address these issues, we propose a new method.

3 PROPOSED FRAMEWORK

Our framework is shown in Fig. 2 with a multi-scale feature extraction module (MFEM) and a joint attention module (JAM). The raw frames of the input video are first fed into the proposed MFEM module to obtain the pyramid-level features in multi-scales. Then, the textual features are extracted by GLoVe and LSTM from the word sequence in natural language. Next, features of two modalities are fused by JAM, whose parameters are shared for feature map sequences with multiple scales. In JAM, the spatial features in the video are firstly coupled with the temporal dimension and then relate to the textual data. Finally, according to two different question forms (multi-choice or open-ed), the answer is generated by the classification or regression branch.

3.1 Multi-scale Feature Extraction Module

We detail our visual feature extraction module in MJAN in this section. To the best of our knowledge, it is the first attempt that uses features extracted from the detection network. The parameters and datasets about video pre-processing are given in the 'Experiments' section.

Recent models for VideoQA tasks use a pre-trained classification network to extract visual features, such as ResNet-152 [14] or C3D [32]. In those methods, the extracted feature from the input can be represented by $v \in \mathbb{R}^{H \times W \times C_v}$. H and W respectively denote the height and width of the feature map, which are set as 7 for the input frame with the size of 224×224.

Fig. 3 shows our MFEM. We employ the RetinaNet [26], a detection network to extract visual features, which is pre-trained on VrR-VG dataset [25] and maintains constant during training. For a single frame, the extracted feature can be represented as $\{P_3, P_4, P_5, P_6, P_7\}$, where P_3 to P_7 are the pyramid level features. As in [26], $P_i \in \mathbb{R}^{H_i \times W_i \times C_v}$ has a resolution 2^i lower than the input, and the dimension of all the features is set to $C_v = 256$. Therefore, for the frame with size 224×224, the feature map scales of H_3 to H_7 are 28, 14, 7, 4, 2, respectively. W_3 to W_7 have the same scales. Finally, for the i^{th} pyramid level, the visual feature sequence can be expressed as $V_i = \{P_i^1, P_i^2, ..., P_i^N\}$, with N as the length of the video.

Classification networks are usually chosen as the backbone in the ImageQA and VideoQA task [2, 8, 17, 35]. As the classification task only focuses on a single large-scale object in the image, these models are inefficient due to the lack of multiple-scale information in the representation. On the contrary, to some extent, the detection model can more comprehensively represent each visual object in the image with more locally detailed information. Considering that the question is often related to the spatio-temporal context information in VideoQA, our method should be more effective and suitable than conventional methods [16].

Unlike [16] which is based on the high-confidence object of the detection network, we take the detection network as the backbone and achieve more robustness. Even if an object is not marked as a target during pre-training, it can be effectively characterized by the feature map used in the feature extraction process as a visually significant object. Furthermore, the pyramid-level features ensure that the visual object can be effectively represented regardless of its scales.

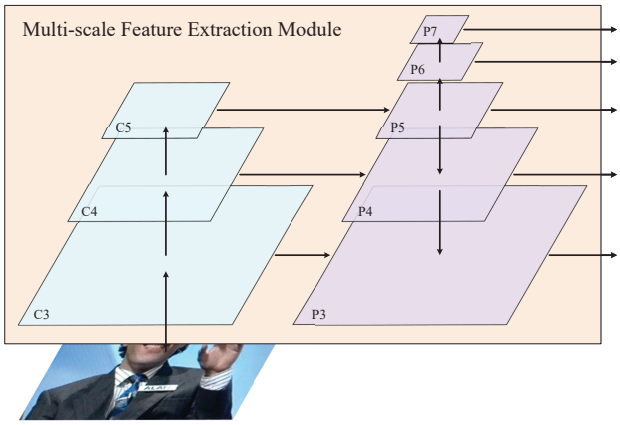

Figure 3: The architecture of the Multi-scale Feature Extraction Module from a pre-trained RetinaNet. As in [26], C_i and P_i have resolution 2^i lower than the input. The dimension of all the output features is set to 256.

3.2 Textual Understanding

To further improve the representation ability we introduce a textual feature extraction method in our MJAN. In the VideoQA task, questions are in the form of natural language. We employ GLoVe [28] pre-trained on the Common Crawl dataset to extract the embedding feature for each word. Then, the embedding features are input to LSTM to obtain the textual features of the whole sentence. This process can be expressed as:

$$e_i = \text{GLoVe}(w_i), \tag{1}$$

$$t_i = \text{LSTM}_{text}(e_i, l_{i-1}), \tag{2}$$

where w_i, e_i, t_i denote the one-hot vector, embedding features, and LSTM state of the i^{th} word, respectively. The last state $t_M \in \mathbb{R}^{C_t}$ is taken as the output of this module, where M is the number of the words in the question. And the dimension of the LSTM hidden state is $C_t = 256$.

3.3 Joint Attention Module

We introduce a joint attention module (JAM) to calculate attention weights based on both temporal and spatial dimensions, as shown in Fig. 4.

Jointly calculating spatial and temporal attention matters. Calculating them separately will degrade the performance of the model. A very common scenario is that the visual target involved in the question may not appear in a certain period in the video, but these irrelevant frames will also be input into the spatial attention module. The normalization operation in the attention is bound to cause it to "pay attention" to certain visual areas and thus cannot ignore that frame. This unrelated and wrong visual information will harm the subsequent timing analysis. Jang *et al.* [17] confirmed the aforementioned result in their work.

The function of the attention mechanism is to embed the textual information guidance in the model to omit the ineffective visual information. Intuitively, the attention mechanism enables the model to find relevant clips or regions in the video. The input feature map sequence $V_i \in \mathbb{R}^{N\times H_i\times W_i\times C_v}$ can be regarded as composed of $H_i \times W_i$ feature sequences, each of which has a length of T.

Bi-LSTM is employed to obtain temporal information for each feature sequence. Then, two 3D convolutional layers with *Relu*, whose kernel size is $3 \times 3 \times 3$, are used to further extract the local spatial-temporal neighborhood information and output $V_i' \in \mathbb{R}^{N\times H_i\times W_i\times C_v}$. The textual feature t_M is repeated and concatenated to each position of V_i'. The concatenated feature is input to the linear part, which consists of two fully connected layers, to obtain the weight $\alpha_i \in \mathbb{R}^{N\times H_i\times W_i\times 1}$ of each position. Finally, the original feature sequence is summed according to the weight of α_i. The above process can be expressed as:

$$V_i' = Subnet(V_i), \tag{3}$$

$$t_M' = Repeat(t_M), \tag{4}$$

$$\alpha_i = Softmax(W_2 Tanh(W_1([V_i', t_M']) + b_1) + b_2), \tag{5}$$

$$\tilde{V}_i = \sum_{x=1}^{H_k}\sum_{y=1}^{W_k}\sum_{t=1}^{N} V_i^{x,y,t}\alpha_i^{x,y,t}, \tag{6}$$

where $\tilde{V}_i \in \mathbb{R}^{C_v}$ is the output of our JAM model. W_1, W_2, b_1, b_2 are the learnable parameters in two fully connected layers. $Tanh(.)$ and $Softmax(.)$ are activation functions of neurons.

In [17, 18], the spatial attention is calculated first, and then the temporal attention is calculated, while JAM is the first attempt to calculate the two attentions jointly. If there is no question related to visual information in frames, they should be ignored by the temporal attention mechanism first. Based on the motivation, our method can make a joint analysis of features from the temporal and spatial dimensions. According to the three-dimensional neighborhood information, JAM calculates the correlation between the textual feature and visual feature at the different spatial-temporal dimensions.

3.4 The Answer Decoder Module

As shown in Fig. 2, the visual features $\tilde{V}_i$ in each level and textual features t_M are weighted and concatenated together. Our model is divided into regression and a classification branch to consider multi-choice and open-ended questions. Both of them consist of two fully connected layers. The final output depends on the question form.

For the multi-choice question, the model uses the regression branch to generate the confidence of the candidate answer. The last fully-connected layer can be expressed as:

$$y = Wx. \tag{7}$$

For the open-ended question, the model uses a classification branch to infer the answer. The output is defined as:

$$y = Softmax(Wx + b). \tag{8}$$

Specifically, although the question of 'How many' belongs to the open-ended form, we can infer a priori that the answer is an integer. Therefore, the regression branch is used, and the output is

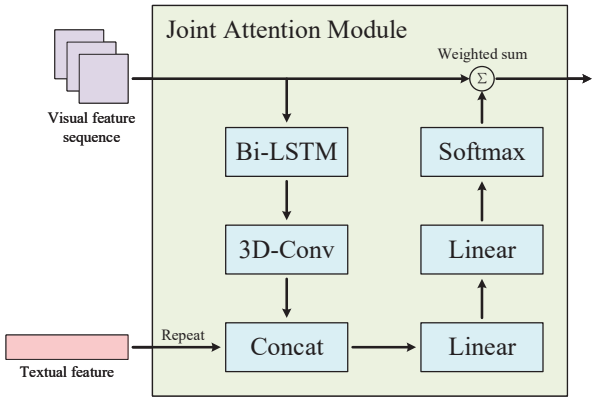

Figure 4: The architecture of the Joint Attention Module that analyzes the features in temporal and spatial dimensions together. The textual feature is repeated to fit the size of the visual feature sequence.

Task	Train	Test	Total
Count	26,843	3,554	30,397
Frame	39,392	13,691	53,083
Action	20,475	2,274	22,749
Trans	52,704	6,232	58,936
Total	139,414	25,751	165,165

Table 1: Statistics of the TGIF-QA dataset. The number of training and testing samples are listed. The Count and Frame are open-ended, while the Action and Trans are multi-choice with five answer candidates.

expressed as:

$$y = [Wx + b], \tag{9}$$

where [.] denotes the rounding operation.

4 EXPERIMENTS

In this section, the proposed model Multi-scale Joint Attention Network (MJAN) is evaluated on three datasets: TGIF-QA [17], MSVD-QA [33], and MSRVTT-QA [33]. First, we report the training details in experiments. Then the experimental results compared with the state-of-the-art models and the ablation studies on TGIF-QA are presented. At last, to further verify the performance of the proposed network, we conduct the experiments on the MSVD-QA and MSRVTT-QA datasets.

4.1 Implementation Details

For each video, 32 equally spaced samples in the temporal dimension are taken as network input. For the video with a length of fewer than 32 frames, the method of repeating the last frame is used to supply the samples. Each frame is reduced to 256×256 size by the bilinear interpolation. Then the 224×224-sized region is cropped as the input of the model. As a method of data augmentation [15], the cropped region of each video is random, while the cropped region of different frames of the same video needs to be consistent.

Item	Specification
CPU	Intel(R) Core(TM) i7-6850K CPU @ 3.60GHz
Memory	64GB
GPU	NVIDIA TITAN Xp
GPU Memory	12GB

Table 2: Hardware used in training MJAN.

Item	Version
Ubuntu	16.04
Python	3.5.1
Tensorflow	2.0.0
CUDA	10.0
Opencv	4.0.0
Numpy	1.17.2

Table 3: Software used in training MJAN.

In our MJAN, the visual backbone RetinaNet [26] is pre-trained on the VrR-VG dataset [25]. The word embedding method GLoVe is pre-trained on Common Crawl. All the other parameters are initialized by the Xavier method [10]. The batch size is 32, and the optimization is Adam. The maximum learning rate is $1e^{-3}$, while the minimum learning rate is $1e^{-5}$. For each task, the model has been trained 10 epochs, and the learning rate is decreased by cosine decay.

4.2 Experiment Environment

The hardware and software equipment used in training the proposed MJAN is shown in Tab. 2 and Tab. 3, respectively.

4.3 Experiments on TGIF-QA

4.3.1 TGIF-QA. TGIF-QA is a large dataset in VideoQA task [17]. It contains about $71k$ animated GIFs from the Tumblr GIFs dataset [24] and about $165k$ question-answer pairs. According to the different contents of the question, the experiments for the TGIF-QA dataset

2*Model	Count	Frame	Action	Trans
	MSE	Accuracy(%)		
ST(R+C)	4.38	48.2	60.1	65.7
ST-Sp(R+C)	4.28	45.5	57.3	63.7
ST-Tp(R+C)	4.40	49.3	60.8	67.1
ST-SpTp(R+C)	4.56	47.8	57.0	59.6
HME(R+C)	4.02	53.8	73.9	77.8
HGA(R+C)	4.09	55.1	75.4	81.0
ST-Tp(R+F)	4.32	49.5	62.9	69.4
Co-mem(R+F)	4.10	51.5	68.2	74.3
L-GCN(R+M)	3.95	56.3	74.3	81.1
HCRN(R+X)	**3.82**	55.9	75.0	81.4
CT-SAN(R)	5.13	39.6	56.1	64.0
STA(R)	4.25	56.6	72.3	79.0
PSAC(R)	4.27	55.7	70.4	76.9
QueST(R)	4.19	**59.7**	75.9	81.0
MJAN(Ret)	3.92	58.2	**77.4**	**82.3**

Table 4: Comparison with the state-of-the-art methods on the TGIF-QA dataset. The models are grouped according to different visual feature extractors. R, C, F, M, X refer to ResNet [14], C3D [32], Optical Flow, Mask R-CNN [13], and 3D-ResNeXt [12] features, respectively. The proposed model MJAN employs RetinaNet to obtain visual features. Compared with other models, including ST [17], HME [5], HGA [19], Co-memory [7], L-GCN [16], HCRN [20], CT-SAN [36], STA [8], PSAC [23], QueST [18], our model achieves very competitive results.

2*Backbone	2*Feature	Count	Frame	Action	Trans
		MSE	Accuracy(%)		
ResNet-50, ImageNet pre-training	C_5	4.26	55.9	73.4	80.8
2*RetinaNet-50, VrR-VG pre-training	P_5	4.18	55.5	76.0	81.2
	P_3-P_7	3.92	58.2	77.4	82.3

Table 5: Ablation studies on the MFEM

2*Model	Count	Frame	Action	Trans
	MSE	Accuracy(%)		
MJAN w/o JAM	4.37	52.4	72.2	77.7
MJAN	3.92	58.2	77.4	82.3

Table 6: Ablation studies on the JAM.

can be divided into four tasks: Count, Frame, Action, and Transition. The number of samples used for training and test in each task is shown in Tab. 1.

Repetition Count (Count): It is about the number of repetitions of action, such as 'How many times does the man wave his hand?'. The question has an open-ended form, while the answer range is from 0 to 10.

Repeating Action (Action): It is about the action category, such as 'What does the man do two times?'. The question has a multi-choice form, and the dataset provides five alternative answers.

State Transition (Trans): It is about the sequence of actions, such as 'What does the man do before walking away?'. The question has a multi-choice form, and the dataset provides five alternative answers.

Frame QA (Frame): It is about the static visual information, such as color or number of objects, which can be obtained from a single frame. For instance, 'What is the color of the hair?'. The question has an open-ended form.

4.3.2 Evaluation Metrics and Loss Functions. For the Count task, the regression branch is used, and the output is rounded as the final answer. Mean Square Error (MSE) is taken as both an evaluation metric and a loss function for the Count task.

For two multi-choice tasks (Action and Trans), one candidate answer is concatenated to the question so that it can be considered as a part of the network input. Then, the confidence of the single candidate answer is generated by the regression branch. We adopt a hinge loss to update the parameters and use the accuracy to evaluate the performance.

	What	Who	How	When	Where
Train	19,485	10,479	736	161	72
Val	3,995	2,168	185	51	16
Test	8,149	4,552	370	58	28
Total	31,629	17,199	1,291	270	116

Table 7: Statistics of the MSVD-QA dataset. The number of training and testing samples are listed. All the questions are open-ended.

	What	Who	How	When	Where
Train	108,792	43,592	4,067	1,626	504
Val	8,337	3,439	344	106	52
Test	49,869	20,385	1,640	677	250
Total	166,998	67,416	6,051	2,409	806

Table 8: Statistics of the MSRVTT-QA dataset. The number of training and testing samples are listed. All the questions are open-ended.

For the Frame task, we selected 1000 answers with the highest frequency in the dataset, which means it is treated as a multi-class classification task. The loss function is cross-entropy, and the evaluation metric is accuracy.

4.3.3 Comparisons with the State-of-the-art Model. The results of the proposed MJAN are compared with recent state-of-the-art methods, as shown in Tab. 4. The models are grouped according to different visual feature extractors. R, C, F, M, X refer to ResNet [14], C3D [32], Optical Flow, Mask R-CNN [13], and 3D-ResNeXt [12] features, respectively. We employed RetinaNet [26] to obtain visual features. The results show that the proposed MJAN outperforms the best methods by 1.5% and 0.7% of accuracy for Action and Trans tasks. For Count and Frame tasks, our model just 0.1 of MSE and 1.5% of accuracy behind the state-of-the-art methods (HCRN [20] and QueST [18]).

It is noted that we only use the RetinaNet features with a similar computational cost as ResNet, but achieve comparable results for multiple backbone methods (ST [17], HME [5], HGA [19], Co-memory [7], L-GCN [16], HCRN [20]). The differences between ResNet and RetinaNet are discussed in the ablation study.

4.3.4 Ablation Studies. To further evaluate the effectiveness of the pioneering contributions of MJAN, a series of ablation studies are conducted on the TGIF-QA dataset. Tab. 5 shows the ablation experiment results on the MFEM. First, we change the backbone to the ResNet-50 pre-trained on ImageNet and use the C_5 (the output of Conv4_3) [14] as the visual feature, which is a common design of previous research work [17]. Then, the RetinaNet pre-trained on VrR-VG is employed as the backbone, and only P_5 is used as the feature map, the size of which is the same as the C_5. Finally, the MJAN model with full multi-scale features from P_3 to P_7 is evaluated. The results show that although only a single scale feature is used, the model based on detection is superior to the method based on the classification in Count, Action, and Trans tasks. For the Frame task, using P_5 is just 0.4% behind using C_5. Then, as expected, the performance of the network is significantly improved by employing multi-scale information.

Tab. 6 shows the ablation experiment results on JAM. In experiments (MJAN w/o JAM), the $\tilde{V}_i$ is calculated by taking the mean value in spatial and temporal dimensions. Undoubtedly, the attention mechanism has greatly improved the performance of the model.

Model	What	Who	How	When	Where	All
ST-VQA	18.1	50.0	**83.8**	72.4	28.6	31.3
Co-Mem	19.6	48.7	81.6	**74.1**	31.7	31.7
GRA	20.6	47.5	83.5	72.4	**53.6**	32.0
HME	22.4	50.1	73.0	70.7	42.9	33.7
QueST	24.5	52.9	79.1	72.4	50.0	36.1
HGA	23.5	50.4	83.0	72.4	46.4	34.7
L-GCN	-	-	-	-	-	34.3
MJAN	**24.9**	**53.8**	79.7	72.4	46.4	**36.7**

Table 9: Comparison with the state-of-the-art methods on the MSVD-QA. L-GCN [16] only reports the accuracy of all test samples.

Model	What	Who	How	When	Where	All
ST-VQA	24.5	41.2	78.0	76.5	34.9	30.9
Co-Mem	23.9	42.5	74.1	69.0	42.9	32.0
GRA	26.2	43.0	80.2	72.5	30.0	32.5
HME	26.5	43.6	82.4	76.0	28.6	33.0
QueST	27.9	45.6	83.0	75.7	31.6	34.6
HGA	29.2	45.7	83.5	75.2	34.0	35.5
MJAN	**29.8**	**46.8**	**83.8**	**79.0**	**44.0**	**36.3**

Table 10: Comparison with the state-of-the-art methods on the MSRVTT-QA.

4.4 Experiments on MSVD-QA and MSRVTT-QA

To further evaluate the performance of the MJAN, we tested on MSVD-QA and MSRVTT-QA. Both datasets are published by Xu *et al.* in [33]. They are generated from existing video description datasets [3, 34]. The questions of the two datasets are open-ended and can be divided into 5 types: 'What', 'Who', 'How', 'When', and 'Where'. Tab. 7 and Tab. 8 show the statistic of MSVD-QA and MSRVTT-QA, respectively. Similar to the Frame task in TGIF-QA, 1000 answers with the highest frequency are selected, which means it is treated as a multi-class classification task.

Tab. 9 shows the results compared with the recent model on the MSVD-QA. Our model outperforms the best methods by 0.4% and 0.9% of accuracy for 'What' and 'Who' tasks. On the other three tasks ('How', 'When' and 'Where'), our model is not as good as the best results, but the results are still competitive. Due to the high proportion of the two types of questions ('what' and 'who') accounted for more than 96% of the total number of test samples in the whole dataset, the overall performance of our model is also 0.6% better than the best SOTA results. Tab. 10 shows the results

compared with the recent model on the MSRVTT-QA. On this larger dataset, our model outperforms the current best model in all tasks.

5 CONCLUSION

This work presents a novel Multi-scale Joint Attention Network (MJAN) model for video question answering. It contains an innovative Multi-scale Feature Extraction Module (MFEM) and an innovative Joint Attention Module (JAM). MFEM can comprehensively and accurately extract multi-scale visual features. JAM can represent effective visual information with question guidance in the spatial and temporal dimensions jointly. Compared with other state-of-the-art methods, our experimental results show the superiority of the proposed MJAN on the TGIF-QA, MSVD-QA, and MSRVTT-QA and datasets.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6077–6086.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*. 2425–2433.

[3] David Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 190–200.

[4] Long Hoang Dang, Thao Minh Le, Vuong Le, and Truyen Tran. 2021. Object-Centric Representation Learning for Video Question Answering. arXiv:2104.05166 [cs.CV]

[5] Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. 2019. Heterogeneous memory enhanced multimodal attention model for video question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1999–2007.

[6] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In *Conference on Empirical Methods in Natural Language Processing*. 457–468.

[7] Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. 2018. Motion-appearance co-memory networks for video question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6576–6585.

[8] Lianli Gao, Pengpeng Zeng, Jingkuan Song, Yuan-Fang Li, Wu Liu, Tao Mei, and Heng Tao Shen. 2019. Structured Two-Stream Attention Network for Video Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 6391–6398.

[9] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. 2016. Compact bilinear pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 317–326.

[10] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth International Conference on Artificial Intelligence and Statistics*. 249–256.

[11] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6904–6913.

[12] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2018. Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet?. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6546–6555.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*. 2961–2969.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[15] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. 2019. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 558–567.

[16] Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. 2020. Location-Aware Graph Convolutional Networks for Video Question Answering.. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 11021–11028.

[17] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. 2017. TGIF-QA: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2758–2766.

[18] Jianwen Jiang, Ziqiang Chen, Haojie Lin, Xibin Zhao, and Yue Gao. 2020. Divide and Conquer: Question-Guided Spatio-Temporal Contextual Attention for Video Question Answering.. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 11101–11108.

[19] Pin Jiang and Yahong Han. 2020. Reasoning with Heterogeneous Graph Alignment for Video Question Answering.. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 11109–11116.

[20] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. 2020. Hierarchical Conditional Relation Networks for Video Question Answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9972–9981.

[21] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. 2018. TVQA: Localized, Compositional Video Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 1369–1379.

[22] Xiangpeng Li, Lianli Gao, Xuanhan Wang, Wu Liu, Xing Xu, Heng Tao Shen, and Jingkuan Song. 2019. Learnable Aggregating Net with Diversity Learning for Video Question Answering. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1166–1174. https://doi.org/10.1145/3343031.3350971

[23] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. 2019. Beyond rnns: Positional self-attention with co-attention for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 8658–8665.

[24] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. 2016. TGIF: A new dataset and benchmark on animated GIF description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4641–4650.

[25] Yuanzhi Liang, Yalong Bai, Wei Zhang, Xueming Qian, Li Zhu, and Tao Mei. 2019. VrR-VG: Refocusing visually-relevant relationships. In *Proceedings of the IEEE International Conference on Computer Vision*. 10403–10412.

[26] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*. 2980–2988.

[27] Fei Liu, Jing Liu, Richang Hong, and Hanqing Lu. 2019. Erasing-Based Attention Learning for Visual Question Answering. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1175–1183. https://doi.org/10.1145/3343031.3350993

[28] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1532–1543.

[29] Mengye Ren, Ryan Kiros, and Richard Zemel. 2015. Exploring models and data for image question answering. In *Advances in Neural Information Processing Systems*. 2953–2961.

[30] Xiaomeng Song, Yucheng Shi, Xin Chen, and Yahong Han. 2018. Explore Multi-Step Reasoning in Video Question Answering. In *Proceedings of the 26th ACM International Conference on Multimedia* (Seoul, Republic of Korea) *(MM '18)*. Association for Computing Machinery, New York, NY, USA, 239–247. https://doi.org/10.1145/3240508.3240563

[31] Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. 2018. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4223–4232.

[32] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 4489–4497.

[33] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. 2017. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM International Conference on Multimedia*. 1645–1653.

[34] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5288–5296.

[35] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 21–29.

[36] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3165–3173.

[37] Yueting Zhuang, Dejing Xu, Xin Yan, Wenzhuo Cheng, Zhou Zhao, Shiliang Pu, and Jun Xiao. 2020. Multichannel Attention Refinement for Video Question Answering. *ACM Trans. Multimedia Comput. Commun. Appl.* 16, 1s, Article 24 (March 2020), 23 pages. https://doi.org/10.1145/3366710