Object-Centric Representation Learning for Video Question Answering

Long Hoang Dang, Thao Minh Le, Vuong Le, Truyen Tran

Applied Artificial Intelligence Institute, Deakin University, Australia

{hldang,lethao,vuong.le,truyen.tran}@deakin.edu.au

Abstract—Video question answering (Video QA) presents a powerful testbed for human-like intelligent behaviors. The task demands new capabilities to integrate video processing, language understanding, binding abstract linguistic concepts to concrete visual artifacts, and deliberative reasoning over space-time. Neural networks offer a promising approach to reach this potential through learning from examples rather than handcrafting features and rules. However, neural networks are predominantly feature-based – they map data to unstructured vectorial representation and thus can fall into the trap of exploiting shortcuts through surface statistics instead of true systematic reasoning seen in symbolic systems. To tackle this issue, we advocate for object-centric representation as a basis for constructing spatio-temporal structures from videos, essentially bridging the semantic gap between low-level pattern recognition and high-level symbolic algebra. To this end, we propose a new query-guided representation framework to turn a video into an evolving relational graph of objects, whose features and interactions are dynamically and conditionally inferred. The object lives are then summarized into résumés, lending naturally for deliberative relational reasoning that produces an answer to the query. The framework is evaluated on major Video QA datasets, demonstrating clear benefits of the object-centric approach to video reasoning.

I. INTRODUCTION

Answering natural questions regarding a video suggests the capability to purposefully reason about what we see in a dynamic scene. Modern neural networks promise a scalable approach to train such reasoning systems directly from examples in the form (video, question, answer). However, the networks' high degree of trainability leads to an undesirable behavior: They tend to exploit shallow patterns and thus creating shortcuts through surface statistics instead of true systematic reasoning. Stretching this methodology leads to data inefficiency, poor deliberative reasoning, and limited systematic generalization to novel questions and scenes [6].

Humans take a different approach. Since early ages, we identify objects as the core "living" construct that lends itself naturally to high-level whole scene reasoning of object compositions, temporal dynamics, and interaction across space-time [27]. Objects admit spatio-temporal principles of cohesion, continuity in space-time, and local interactions, allowing humans to infer the past and predict the future without relying too much on constant sensing. Cognitively, objects offer a basis for important abstract concepts such as sets and numbers, which can be symbolically manipulated without grounding to sensory modalities; and concrete concepts such as spatial layout and affordances. Mathematically, objects offer a decompositional
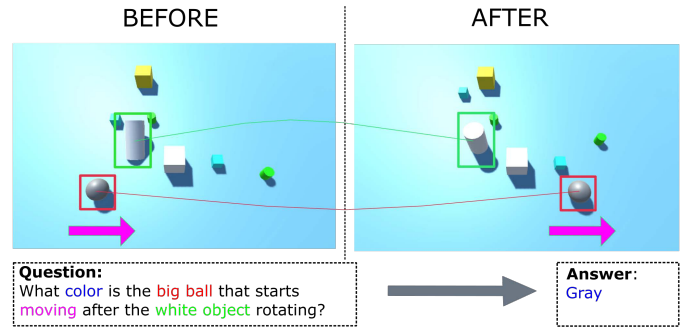

Figure 1. The *big ball* can have two overall temporal parts, one before *moving*, and the other after. To predict a correct answer, the model needs to localize the action (*moving*), understand the evolution of the object (*the big ball*) and capture the contextualized interaction with its neighbors (*after the write object rotating*).

scheme to disentangle the complexity of the world, giving rise to modularity through encapsulation, separating sensory patterns from functions, thus enabling high-order transfer learning across completely separate domains.

Inspired by these capabilities, we advocate for new paths to train neural networks for video question answering (Video QA) via *object-centric video representation* [2]. Here objects in video are primary constructs that have unique evolving lives throughout space-time. In addition to usual visual parts, moving objects have temporal parts [8], and these are essential to understand its evolution and contextualized interaction with other objects. More concretely, we first extract from a video as a set of object tubelets using recent deep neural nets for object detection and tracking. Tubelets are then partitioned into short sub-tubelets, each of which corresponds to a brief period of object life with small change in appearance and position. This allows object representations to be temporally summarized and refined, keeping only information relevant to the query. Objects living in the same period form context-specific relationships, which are conditionally inferred from the scene under the guidance of the query. The objects and their relationships are represented as a graph, one per period. As the objects change throughout the video, their relationships also evolve accordingly. Thus the video is abstracted as a *query-conditioned evolving object graph*.

The object graphs are then parameterized as a sequence of deep graph convolutional networks (DGCNs) [16]. The DGCNs refine the object's temporal part representations within each
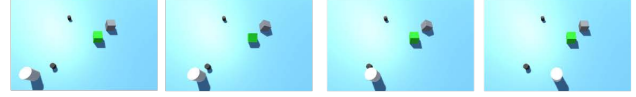
Question: Who is demonstrating his skills with a sword in front of a crown?
Type: Who (MSVD-QA)
Ground truth: man

Question: What are a group of men and women doing??
Type: What (MSRVTT-QA)
Ground truth: dance
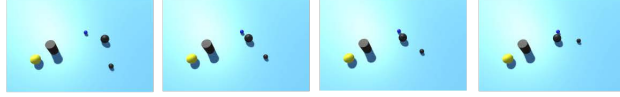
Question: How many small balls that have the same action type as the big black ball?   **Type:** Count (SVQA)
Ground truth: 1

Question: Are there any big gray cubes rotating that are on the right side of the big green object rotating?   **Type:** Exist (SVQA)
Ground truth: yes

Figure 2. Several examples of scenes and question-answer pairs in Video QA datasets with a wide range of question types. It is clear to see that Video QA models must understand the compositions, temporal dynamics, and interaction across space-time of objects in a whole video to provide the correct answer.

short period. The representation serves as input to BiLSTM, which sequentially connects different temporal parts of the same object. Thus the first and the last states of a BiLSTM effectively encode the lifetime information of an object in the context of others, i.e., a contextualized résumé. At this stage, the video is abstracted into a set of résumés which are then reasoned about given a linguistic query using any general-purpose relational reasoning engine.

The system is evaluated on three major Video QA datasets: MSVD-QA [30] consisting of 50K questions with 2K videos, MSRVTT-QA [31] with 243K questions over 10K videos, and SVQA [26] with 120K questions over 12K videos of moving objects. These datasets are suitable to test the reasoning capability against complex compositional questions about spatial-temporal relationships among objects. Our results establish new state-of-the-art accuracies on the datasets.

To summarize, we make the following contributions: (a) proposal of a new object-centric representation for Video QA, (b) introduction of a new dynamic graph neural architecture that enables learning to reason over multiple objects, their relations and events, and (c) establishing new state-of-the-art results on three common Video QA benchmarks of MSVD-QA, MSRVTT-QA, and SVQA.

II. RELATED WORK

Question answering in its fullest form requires a strong reasoning capability for deliberately manipulating previously acquired knowledge to form the answer of a given question [1]. While high-level reasoning in humans seems to involve symbols and logic [21], the computational substrate is largely neuronal, suggesting that there exists an intermediate form of the reasoning process that is not yet formally logical, but still powerful [1]. Neural networks can be a good computational model for reasoning [6], [18], but networks need to be dynamic and deliberative as driven by the question, stepping away from simple associative responses. The dynamics can be manifested in many ways: By rearranging computational units on-the-fly [10], by constructing relations as they emerge and linking them

to the answer [18], or by dynamically building neural programs from memories [17].

Visual reasoning presents a great challenge because the formal knowledge needed for reasoning has not been acquired, but must be inferred from low-level pixels. Adding to this challenge, visual question answering typically involves high-level symbolic cues presented in the form of linguistic questions, thus requiring automatic binding between linguistic symbols and visual concepts [10], [18]. A powerful way to bridge this semantic gap is through *object-centric representation* as an abstraction over pixels [6], [18]. Each object has the potential to package object instance features such as its spatio-temporal information, shape, and color, which can solve the significant problem of disentangling complex scenes into manageable units. Learning objects and relations have been demonstrated to be essential for symbolic representations [5]. A popular way to construct objects is through semantic segmentation or object detection. Extending to video requires tracking algorithms which exploit the temporal permanence of object to form tubelets [14].

In recent years *video question answering* (Video QA) has become a major playground for spatio-temporal representation and visual-linguistic integration. A popular approach treats video as a sequential data structure to be queried upon. Video is typically processed using classic techniques such as RNN [34] or 3D CNN [23], augmented by attention mechanisms [32], or manipulated in a memory network [15]. Stepping away from this flat structure, video can be abstracted as a hierarchy of temporal relations [19], [20], which can either directly generate answers given the query, or serve as a representation scheme to be reasoned about by a generic reasoning engine [20]. More recently, objects have been suggested to be the core component of Video QA [33] as they offer clean structural semantics compared to whole-frame unstructured features. Our work pushes along this line of object-centric representation for Video QA, built on the premise that object interactions are local, and their representation should depend on other objects in context as well as being guided by the query. In particular, objects
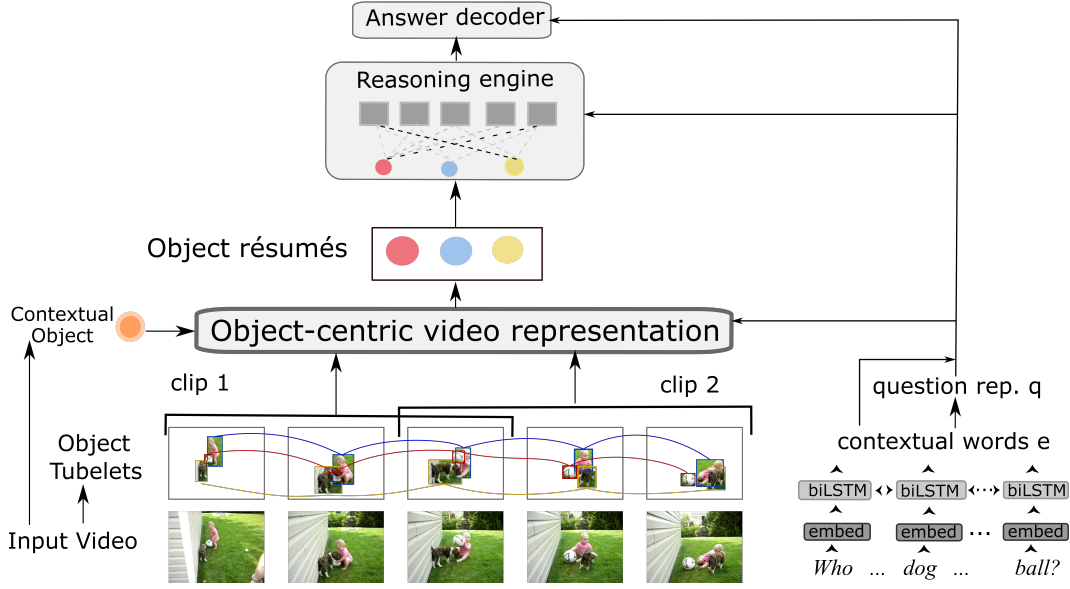
Figure 3. The overall architecture of our proposed model. We represent a *object-centric video representation* where a video is abstracted as a dynamic set of interacting objects living in space-time. The object-centric video representation takes as input tubelets of objects tracked through time and the linguistic query to model the interactions of objects under the modulation of the linguistic interpretation. Output of the object-centric video representation is a set of object résumés. See Fig. 4 for detailed design of this module. These object résumés then serves as the knowledge base for a general-purpose reasoning engine to retrieve the relevant visual information to the query. Finally, an answer decoder translate the presentation to an answer prediction.

are tracked into tubelets so that their lifelines are well defined. Their attributes and relations along such lifeline are constantly updated in the dynamic graph of their social circles of co-occurred fellow objects.

III. METHOD

The Video QA task seeks to build a conditional model P_θ to infer an answer a from of set A, given a video V and a natural linguistic query q:

$$\bar{a} = \arg\max_{a \in A} P_\theta(a \mid q, V). \tag{1}$$

The challenges stem from (a) the long-range dependencies intra–objects across time, and inter–objects across space in the video V; (b) the arbitrary expression of the linguistic question q; and (c) the spatio-temporal reasoning over the visual domain as guided by the linguistic semantics. Fig. 2 gives some typical examples that can describe the major problems in Video QA task. In what follows, we present our object-oriented solution to tackle these challenges.

A. Model Overview

Fig. 3 illustrates the overall architecture of our model. Our main contribution is the object-centric video representation for Video QA in which a video is abstracted as a dynamic set of interacting objects that live in space-time. The interactions between objects are interpreted in the context information given by the query (See Sec. III-B). The output of the object-centric video representation is a set of object résumés. These résumés later serve as a knowledge base for a general-purpose relational reasoning engine to extract the relevant visual information to the question. In the reasoning process, it is desirable that query-specific object relations are discovered and query words

are bound to objects in a deliberative manner. The reasoning module can be generic, as long as it can handle the natural querying over the set of résumés. At the end of our model, an answer decoder takes as input the output of the reasoning engine and the question representation to output a vector of probabilities across words in the vocabulary for answer prediction.

Linguistic question representation: We make use of BiLSTM with GloVe word embedding [22] to represent a given question q. In particular, each word in the question is first embedded into a vector of 300 dimensions. A BiLSTM running on top of this vector sequence produces a sequence of state pairs, one for the forward pass and the other for the backward pass. Each state pair becomes a contextual word representation $e_s = \left[\overleftarrow{h_s}, \overrightarrow{h_s}\right]$, $e_s \in \mathbb{R}^d$, where d is the vector length, and $s = 1, ..., S$ where S is the length of the question. The global sequential representation q_g is obtained by combining the two end states of BiLSTM: $\boldsymbol{q}_g = \left[\overleftarrow{h_1}, \overrightarrow{h_S}\right]$, $q_g \in \mathbb{R}^d$. Finally, we integrate q_g with the contextual words e_s by an attention mechanism to output the final question representation:

$$q = \sum_{s=1}^{S} e_s * \text{softmax}_s(w_q^\top(e_s \odot q_g)), \tag{2}$$

where $w_q \in \mathbb{R}^d$ is network learnable weights.

B. Object-centric Video Representation

We represent an object live based on tubelet – the 3D structure of the object in space-time. At each time step, an object is a 2D bounding box with a unique identity assigned by object tracking (Sec. III-B1). We obtain tubelets by simply linking bounding boxes of the same identities throughout the
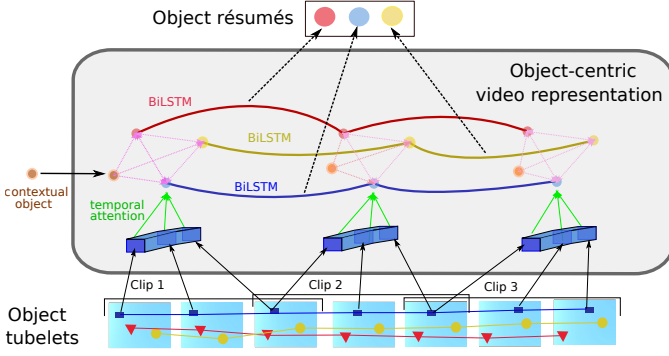
Figure 4. Object-centric Video Representation - Illustration with a scene of three objects. The objects are first detected and tracked (thin matching color lines) from video frames into tubelets (only drawn for the blue rectangle). Their visual and positional features are gathered on each clip with temporal attention (green arrows) into temporal parts (nodes in lower row). The parts of each clip interact by a GCN (dotted purple graph) with the contextual object (orange node) and join temporally by BiLSTMs.

given video. If seeing a video as a composition of events, the interactions between objects often happen in a short period of time. Hence, we can break the object tubelets into sub-tubelets where each sub-tubelet is equivalent to a temporal part [8]. As we model the interactions between objects in the context given by the query, we then summarize each object-wise temporal part into a query-conditioned vector representation (Sec. III-B2). The representations of temporal parts of objects are then treated as nodes of a query-conditioned evolving object graph where the edges of the graph denote the relationships of objects living in the same temporal part. In order to obtain the full representations of object lives through the video, we link object representations in consideration of their interference with neighbor objects across temporal parts. Finally, each object life is summarized into a résumé, preparing the object system for relational reasoning. See Fig. 4 for illustration of our object-centric video representation in a scene of three objects.

1) Constructing Object Tubelets :

Detecting and tracking objects: We use Faster R-CNN [24] to detect frame-wise objects in a given video. This returns, for each object, (a) appearance features $\boldsymbol{v}_a \in \mathbb{R}^{2048}$ (representing "what"); (b) the bounding box $\boldsymbol{v}_b = [x_{min}, y_{min}, x_{max}, y_{max}]$ ("where") and (c) a confidence score. We use a popular object tracking framework DeepSort [29] which makes use of the appearance features and confidence scores of detected bounding boxes to track objects, assigning a unique ID to bounding boxes of the same object. For the ease of implementation, we assume that the objects live from the beginning to the end of video and missing objects at a time are marked with null values. Each object is now a tubelet, consisting of a unique ID, a sequence of coordinates of bounding boxes through time and a set of corresponding appearance features.

Joint encoding of "what" and "where": Positions are of critical importance in reasoning about the spatial arrangement of objects at any given time [35], [28]. Therefore, to represent a geometrical information of each object tubelet at a time step,

we incorporate a spatial vector of 7-dimensions of the bounding boxes, including their four coordinates, width and height, and the size of their relative areas w.r.t the area of the whole video frame, i.e $v_p = \left[\frac{x_{min}}{W}, \frac{y_{min}}{H}, \frac{x_{max}}{W}, \frac{y_{max}}{H}, \frac{\Delta x}{W}, \frac{\Delta y}{H}, \frac{\Delta x \Delta y}{WH} \right]$, where $\Delta x = x_{max} - x_{min}$, $\Delta y = y_{max} - y_{min}$, and (W, H) are the width and height of the video frame, respectively.

Considering a brief period of time where an action takes place, the appearance feature v_a of an object may change very little while the change in its position is more noticeable. For example, given a video where a person is walking towards a car, it is likely that the changes in position is more informative than that in appearance. Hence, we propose to use a multiplicative gating mechanism to encode the *position-specific appearance* of objects

$$\boldsymbol{v}_{ap} = f_1(\boldsymbol{v}_a) \odot g_1(\boldsymbol{v}_p), \qquad (3)$$

where $f_1(\boldsymbol{v}_a)$ is non-linear mapping function and $g_1(\boldsymbol{v}_p) \in (\boldsymbol{0}, \boldsymbol{1})$ is a position gating function to take control the flow of appearance, making the change in representation of objects at different time steps be more discriminated from each other. We choose $f_1(\boldsymbol{v}_a) = \tanh(W_a \boldsymbol{v}_a + b_a)$ and $g_1(\boldsymbol{v}_p) = \text{sigmoid}(W_p \boldsymbol{v}_p + b_p)$ in our later implementation, where W_a and W_p are network parameters.

Besides, in order to compensate for the loss caused by the imperfect object detection, we incorporate a so-called *contextual object* $v_c \in \mathbb{R}^{2048}$ of a video frame. Particularly, we utilize pre-trained ResNet [9] and take pool5 output features as the object feature v_c of each time step.

2) Language-conditioned Representation of Temporal Parts : Once the *position-specific appearance* of objects is computed, we reduce the complexity by grouping several frames into a short clip, creating a language-conditioned temporal part by taking the question into account. The goal is to prepare the parts to be *question–specific*, making them more ready for subsequent relational reasoning.

We split an object life into K equal temporal parts $C = \{C_1, C_2, ..., C_K\}$, where part C_k contains t frames. Let $C_k = \{\boldsymbol{v}_{ap}^{k,1}, .., \boldsymbol{v}_{ap}^{k,i}, ..., \boldsymbol{v}_{ap}^{k,t}\}$, where $\boldsymbol{v}_{ap}^{k,i}$ is the position-conditioned feature computed in Eq. (3) of the i-th frame in the k-th part. Given the question representation $\boldsymbol{q}$, we then apply a *temporal attention* mechanism to compute the probability to which a frame in a part is attended:

$$\alpha_{k,i} = \text{softmax}_i \left(w_s^\top \left((W_q \boldsymbol{q} + b_q) \odot (W_v \boldsymbol{v}_{ap}^{k,i} + b_v) \right) \right), \quad (4)$$

where $\odot$ is the Hadamard product, $\{w_s, W_q, W_v\}$ are learnable weights. The temporal part k of an object is then summarized as:

$$\boldsymbol{c}_k = \lambda * \sum_{i=1}^{t} \alpha_{k,i} \boldsymbol{v}_{ap}^{k,i}, \qquad (5)$$

where λ is a binary mask vector to exclude missed detections of objects. λ is calculated based on indices of the time steps marked with null values as in Sec. III-B1.

3) Query-conditioned Object Graph : Imagining that we have N sequences of objects living together in a period of time (temporal part), their living paths are not only constrained by their own behavior at different points in time but also influenced by their object neighbors living nearby. Besides, in the context of Video QA, relationships between objects are understood in the context given by the query. Note that temporal object parts living in the same period k can be highly correlated, depending on the spatial distance between objects and the semantics of the query. We represent objects in a temporal part k and their relationships by a graph $G_k = (C_k, A_k)$, where $C_k = \{c_{k,o}\}_{o=1}^N$ are nodes and $A_k \in \mathbb{R}^{N \times N}$ is the adjacent matrix. The adjacent matrix A_k is query-dependent and is given by:

$$a_{k,o} = \text{softmax}_o(w_a^\top([\boldsymbol{c}_{k,o}, \boldsymbol{c}_{k,o} \odot q])), \qquad (6)$$
$$A_k = a_k^\top a_k, \qquad (7)$$

where $\boldsymbol{c}_{k,o}$ is representation of part k of object o, as computed in Eq. (5), and a_o is the probability to which object o is attended.

Given the adjacent matrix A_k, we use graph convolutional networks (GCNs) [16] to correlate objects and systematically refine all the object representations by considering their connections with other objects in the same temporal part. We stack GCNs into layers with skip-connections between them, allowing the refinement to run through multiple rounds. To mitigate to effects of imperfect object detection, we propose to use the contextual features of video frames v_c as explained in Sec. III-B1 as an additional node for graph G_k. Different from the original bidirectional graph in [16], our temporal-part-specific object graph G_k is now considered as a heterogeneous graph where connections between objects are bidirectional while the connections between the contextual node and other nodes are directional. Let $H_k^0 = (\boldsymbol{c}_{k,1}, \boldsymbol{c}_{k,2}, ..., \boldsymbol{c}_{k,N}) \in \mathbb{R}^{N \times d}$ be the initial node embedding matrix, we refine representations of nodes in each graph G_k as follows:

$$F_l\left(H_k^{l-1}\right) = W_2^{l-1} g\left(A_k H_k^{l-1} W_1^{l-1} + W_{v_c} v_c^k + b^{l-1}\right), \quad (8)$$

where $l = 1, 2, ...L$ is the number of GCN layers, and $g(\cdot)$ is nonlinear transformation chosen to be ELU in our implementation. The skip-connection is then added between GCN layers to do multiple round representation refinement:

$$H_k^l = g\left(H_k^{l-1} + F_l\left(H_k^{l-1}\right)\right). \qquad (9)$$

Finally, the contextualized parts representation is then $(\tilde{\boldsymbol{c}}_{k,1}, \tilde{\boldsymbol{c}}_{k,2}, ..., \tilde{\boldsymbol{c}}_{k,N}) \leftarrow H_k^L$.

4) Video as Evolving Object Graphs : An object is a sequence of temporal parts whose representation is contextualized in Sec. III-B3. Hence the video is abstracted as a spatio-temporal graph whose spatial and temporal dependencies are conditioned on the query, as illustrated in Fig. 4. Let $\tilde{\boldsymbol{c}}_k$ be contextualized part representation of the object. Temporal parts are then connected through a BiLSTM:

$$\overrightarrow{h}_k = \overrightarrow{\text{LSTM}}\left(\overrightarrow{h}_{k-1}, \tilde{\boldsymbol{c}}_k\right), \ k > 0 \ \text{and} \qquad (10)$$
$$\overleftarrow{h}_k = \overleftarrow{\text{LSTM}}\left(\overleftarrow{h}_{k+1}, \tilde{\boldsymbol{c}}_k\right), \ k < K, \qquad (11)$$

with the first state initialized as $\overrightarrow{h}_0 = \overrightarrow{W}_q \boldsymbol{q} + \overrightarrow{b}_q$ and $\overleftarrow{h}_K = \overleftarrow{W}_q \boldsymbol{q} + \overleftarrow{b}_q$, respectively.

The dynamic object graph represents the scene evolution well but it remains open how to efficiently perform arbitrary reasoning, given the free-form expression of the linguistic query. For example, given a scene of multiple objects of different shapes, we want to answer a question "Is there a cylinder that start rotating before the green cylinder?". For this we need to search for the matching cylinders, and traverse the graph to identify the event of rotating, then follow up the cylinder trajectories before the event. In the process of matching, we need to find the right visual colored cylinder that agree with the linguistic words "cylinder", "green cylinder" and "rotating". For large temporal graphs, it poses a great challenge to learn to perform these discrete graph operations.

To mitigate the potential complexity, we propose to integrate out the temporal dimension of the dynamic object graph, creating an unordered set of N object résumés. This method of temporal integration has been empirically shown to be useful in Video QA [20]. In particular, we compute a résumé for each object by summarizing its lifetime using $\boldsymbol{r} = \left[\overleftarrow{h}_1, \overrightarrow{h}_K\right]$, where $\overleftarrow{h}_1$ and $\overrightarrow{h}_K$ are end states of the BiLSTM using Eqs. (10,11). This representation implicitly codes the appearance, geometric information, temporal relations of an object in space-time, as well as the parts relation between objects living in the same spatio-temporal context.

C. Relational Reasoning

Now we have two sets, one consisting of contextualized visual résumés $R = \{\boldsymbol{r}_o\}_{o=1}^N$, and the other consisting of the query representation $Q = \{q_g, e_1, e_2, ..., e_S\}$. Reasoning, as defined in Eq. (1), amounts to the process of constructing the interactions between these two sets (e.g., see [11]). More precisely, it is the process of manipulating the representation of the visual object set, as guided by the word set. An important manipulation is to dynamically construct *language-guided predicates of object relations*, and chain these predicates in a correct order so that the answer emerges. It is worth to emphasize that our *object-centric video representation* can combine with a wide range of reasoning models. Examples of the generic reasoning engines are the recently introduced MACNet (Memory, Attention, and Composition Network) [11] and LOGNet (Language-binding Object Graph Network) [18]. Although these models were originally devised for static images, its generality allows adaptation to Video QA through object résumés.

Dual-process view : The application of generic iterative relational reasoning engines, for object-oriented Video QA somewhat resembles the dual-process idea proposed in [20]. It disentangles the visual QA system into two sub-processes – one

for domain-specific modeling, and the other for deliberative general-purpose reasoning.

D. Answer Decoder

We follow prior works in Video QA [20], [3], [19] to design an answer decoder of two fully connected layers followed by the softmax function to obtain probabilities of labels for prediction. As we treat Video QA as a multi-class classification, we use cross-entropy loss to train the model.

IV. EXPERIMENTS

A. Datasets

We evaluate our proposed architecture on the three recent benchmarks, namely, MSVD-QA, MSRVTT-QA [30], [31] and SVQA [26].

MSVD-QA contains 1,970 real-world video clips and 50,505 QA pairs of five question types: what, who, how, when, and where, of which 61% of the QA pairs is used for training, 13% for validation and 26% for testing.

MSRVTT-QA consists of 243K QA pairs annotated from over 10K real videos. Similar to the MSVD-QA dataset, the questions are classified into five types: what, who, how, when, and where. The proportions of videos in the training, testing and validation splits are 65/30/5%, respectively.

SVQA is designed specifically for multi-step reasoning in Video QA, similar to the well-known visual reasoning benchmark with static images CLEVR [13]. It contains 12K short synthetic videos and 120K machine-generated compositional and logical question-answer pairs covering five categories: *attribute comparison, count, query, integer comparison* and *exist*. Each question is associated with a question program. SVQA helps mitigate several limitations of the current Video QA datasets such as language bias and the lack of compositional logic structure. Therefore, it serves as an excellent testbed for multi-step reasoning in space-time. We follow prior work [20] to use 70% of videos as the training set, 20% of videos as the test set, and the last 10% for cross-validation.

B. Implementation Details

Unless otherwise stated, each video is segmented into 10 clips of 16 consecutive frames. Faster R-CNN[1] [24] is used to detect frame-wise objects. We take 40 tubelets per video for MSVD-QA and MSRVTT-QA while that of SVQA is 30 simply based on empirical experience. Regarding the language processing, we embed each word in a given question into a vector of 300 dimensions, which are initialized by pre-trained GloVe [22]. The default configuration of our model is with $L = 6$ GCN layers for the query-conditioned object graph (See Sec. III-B3), feature dimensions $d = 512$ in all sub-networks. Regarding the general relational reasoning engine, we use 12 reasoning steps on both MACNet and LOGNet with default other parameters as in [18], [11]. We refer our models as OCRL+LOGNet or OCRL+MACNet, respectively, where OCRL stands for *Object-Centric Representation Learning*. Without mentioning explicitly, we report results of the

[1]https://github.com/airsplay/py-bottom-up-attention

Table I
EXPERIMENTS ON MSVD-QA AND MSRVTT-QA DATASETS.

Model	Test accuracy (%)	
	MSVD-QA	MSRVTT-QA
ST-VQA [12]	31.3	30.9
Co-Mem [4]	31.7	32.0
AMU[30]	32.0	32.5
HME [3]	33.7	33.0
HRA [7]	34.4	35.1
HCRN [19]	36.1	35.6
OCRL+LOGNet	**38.2**	**36.0**

OCRL+LOGNet as it generally produces favorable performance over the OCRL+MACNet.

We train our models with Adam optimizer with an initial learning rate of 10^{-4} and a batch size of 64. To be compatible with related works [20], [26], [19], we use accuracy as evaluation metric for all tasks.

C. Comparison against SOTAs

We compare our proposed method against recent state-of-the-art models on each dataset. **MSVD-QA and MSRVTT-QA datasets:** Results on the MSVD-QA and MSRVTT-QA datasets are presented in Table I. As can be seen, our OCRL+LOGNet model consistently surpasses the all SOTA methods. In particular, our model significantly outperforms the most advanced model HCRN [19] by 2.1 absolute points on the MSVD-QA, whereas it slightly advances the accuracy on the MSRVTT-QA from 35.6 to 36.0. Note that the MSRVTT-QA is greatly bigger than the MSVD-QA and it contains long videos with complex relations between objects, so it might need more tubelets per video than the current implementation of 40 tubelets. As for the contribution of each model's component, we provide further ablation studies on MSVD-QA dataset in Sec. IV-E.

SVQA dataset: Table II shows the comparisons between our model against the SOTA methods. Indeed, our proposed network significantly outperforms all recent SOTA models by a large margin (3.7 points). Specifically, we achieve the best performance in a majority of sub-tasks (11/15), which account for 91% of all question-answer pairs, and slightly underperform in the other four categories. We speculate that this is due the advantages of CRN+MACNet [20] in explicitly encoding high-order temporal relations. Nevertheless, the results clearly demonstrate the necessity of an object-centric approach in handling a challenging problem as Video QA.

D. Object-Centric vs. Grid Representation

In order to understand the effects of the object-centric representation learning (OCRL) in comparison with the popular grid representation used by [20], [19], we use a MACNet [11] as the relational reasoning engine for fair comparison. Fig. 5 presents comparison results on MSVD-QA, MSRVTT-QA, and SVQA datasets between the *OCRL* and non-object feature baselines in [20], [19]. For the first two real datasets, the results confirm that the object-oriented representation

| Methods | Exist | Count | Attribute Comparison | | | | | | | | Query | | | | | All |
			More	Equal	Less	Color	Size	Type	Dir	Shape	Color	Size	Type	Dir	Shape	
SA(S)	51.7	36.3	72.7	54.8	58.6	52.2	53.6	52.7	53.0	52.3	29.0	54.0	55.7	38.1	46.3	43.1
TA	54.6	36.6	73.0	57.3	57.7	53.8	53.4	54.8	55.1	52.4	22.0	54.8	55.5	41.7	42.9	44.2
SA+TA	52.0	38.2	74.3	57.7	61.6	56.0	55.9	53.4	57.5	53.0	23.4	63.3	62.9	43.2	41.7	44.9
STRN	54.0	44.7	72.2	57.8	63.0	56.4	55.3	50.7	50.1	50.0	24.3	59.7	59.3	28.2	44.5	47.6
CRN+MAC	72.8	56.7	**84.5**	**71.7**	**75.9**	70.5	76.2	90.7	75.9	57.2	76.1	92.8	91.0	**87.4**	85.4	75.8
OCRL+MA	77.4	56.7	81.2	64.6	65.0	90.0	93.4	90.1	77.0	93.5	**77.8**	92.9	**91.3**	82.5	**89.5**	77.8
OCRL+LO	**81.7**	**61.5**	83.2	64.9	71.4	**92.7**	**97.2**	**94.6**	**88.8**	**95.7**	75.1	90.9	90.3	82.6	86.8	**79.5**
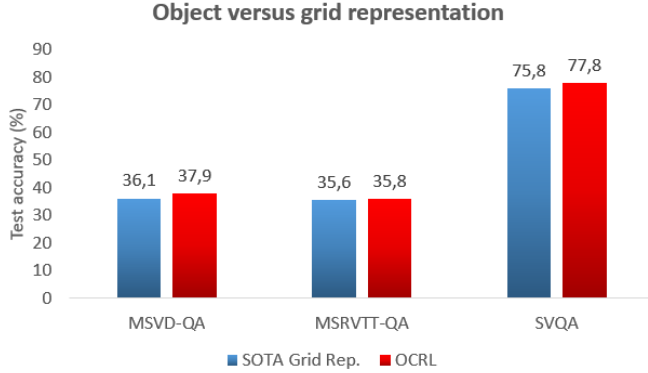
Figure 5. Object versus grid representation on MSVD-QA, MSRVTT-QA, and SVQA datasets. Results for the grid representation is from [19], [20]. For fair comparisons, MACNet is used as the reasoning module for all approaches.

makes it a lot easier to arrive at correct answers than the grid representation counterpart. In particular, the object-centric representation outperforms other SOTA non-object methods on both MSVD-QA and MSRVTT-QA (1.8 and 0.2 points improvement, respectively). For the SVQA, our proposed OCRL model surpasses the grid representation by roughly 2 points. Empirical results on the SVQA specifically reveal the strong desire for proper object-centric representation towards solving multi-step reasoning.

E. Ablation Study

We conduct ablation studies on the MSVD-QA dataset to justify the contributions of different components in our model. Empirically, we have noticed that the contribution of each component is more noticeable when decreasing the number of reasoning steps of the relational module. In particular, we set the number of reasoning steps of LOGNet as 2 in all ablated experiments. Results in Table III reveal that each design component ameliorates the overall performance of the model. The effects are detailed as follows:

Position-specific mechanism: We study the effect of the gating mechanism in the Eq. (3). In particular, we replace this module with a simple concatenation. As can be seen, adding gating mechanism leads to an improvement of 1.1 absolute points. The results convincingly demonstrate the effects of our novel position-gated mechanism comparing to the common approach of combining the appearance and spatial feature [18], [28].

Language-conditioned representation of temporal parts: We study the effect of the language on representing temporal parts in Sec. III-B2. In particular, we replace the temporal attention in Eq. (4) with a simple mean-pooling operation. We experience a significant drop in performance from 37.6% to 35.3% when disregarding the language-condition representation. We conjecture that the *question-specific representations* of object lives make it more ready for the later relational reasoning to arrive at correct answers.

Query-conditioned object graph: We conduct a series of experiments going from shallow GCNs to very deep GCNs to study the effects of the temporal parts representation refinement based on the information carried out by their surrounding objects. Empirical results suggest that it gradually improves the performance when increasing the depth of the GCN, and 6 layers are sufficient on the MSVD-QA dataset.

Object graphs summary: We verify the effect of the sequential modeling of temporal parts as described in Sec. III-B4. The simplest way of computing the summary of object graphs is via an average-pooling operator over all elements in the sequence which is to totally ignore the temporal dependencies between the elements. Results in Table III show that connecting a chain of object temporal parts by a sequence model like BiLSTM is beneficial. Comparing to the clip-based relation network [20], a BiLSTM as in our model is less computationally expensive and easy to implement.

Contextual object: Finally, we investigate the effect of contextual feature in Eq. (8). The performance of our model notably decreases from 37.6 to 35.8 when removing this special object. This result once again confirms the importance of the contextual object in compensating for the loss of information caused by the imperfect object detection.

V. DISCUSSION

We have introduced a novel neural architecture for object-centric representation learning in video question answering. Object-centric representation adds modularity and compositionality to neural networks, offering structural alternatives to the default vectorial representation. This is especially important in

Table III
ABLATION STUDIES ON THE MSVD-QA DATASET. (*) 6 GCN LAYERS.

Model	Validation Acc.
Default config. (*)	37.6
Position-specific mechanism	
w/o Gating mechanism	36.5
Language-conditioned rep. of temporal parts	
w/o Temporal attention	35.3
Interaction graph of temporal parts	
w/ 1 GCN layer	36.9
w/ 4 GCN layers	37.3
w/ 8 GCN layers	37.1
Object graphs summary	
w/o BiLSTM	36.6
Contextual object	
w/o Contextual Obj.	35.8

complex scenes in video with multi-object dynamic interactions. More specifically, our representation framework abstracts a video as an evolving relational graph of objects, whose nodes and edges are conditionally inferred. We also introduce the concept of résumé that summarizes the live of an object over the entire video. This allows seamless plug-and-play with existing reasoning engines that operate on a set of items in response to natural questions. The whole object-centric system is supported by a new dynamic graph neural network, which learns to refine object representation given the query and the context defined by other objects and the global scene. Our architecture establishes new state-of-the-arts on MSVD-QA, MSRVTT-QA, and SVQA – the three well-known video datasets designed for complex compositional questions, relational, spatial and temporal reasoning.

REFERENCES

[1] Léon Bottou. From machine learning to machine reasoning. *Machine Learning*, 94(2):133–149, 2014.

[2] Mikyas T Desta, Larry Chen, and Tomasz Kornuta. Object-based reasoning in VQA. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1814–1823. IEEE, 2018.

[3] Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. Heterogeneous memory enhanced multimodal attention model for video question answering. In *CVPR*, pages 1999–2007, 2019.

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

[5] Marta Garnelo and Murray Shanahan. Reconciling deep learning with symbolic artificial intelligence: representing objects and relations. *Current Opinion in Behavioral Sciences*, 29:17–23, 2019.

[6] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

[7] M. I. Hasan Chowdhury, K. Nguyen, S. Sridharan, and C. Fookes. Hierarchical relational attention for video question answering. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 599–603, 2018.

[8] Katherine Hawley. Temporal parts. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, summer 2020 edition, 2020.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[10] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *ICCV*, pages 804–813. IEEE, 2017.

[11] Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *ICLR*, 2018.

[12] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. *CVPR*, 2017.

[13] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[14] Vicky Kalogeiton, Philippe Weinzaepfel, Vittorio Ferrari, and Cordelia Schmid. Action tubelet detector for spatio-temporal action localization, 2017.

[15] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. DeepStory: video story QA by deep embedded memory networks. In *IJCAI*, pages 2016–2022. AAAI Press, 2017.

[16] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ArXiv*, abs/1609.02907, 2017.

[17] Hung Le, Truyen Tran, and Svetha Venkatesh. Neural stored-program memory. In *ICLR*, 2020.

[18] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Dynamic language binding in relational visual reasoning. In *IJCAI*, 2020.

[19] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9972–9981, 2020.

[20] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Neural reasoning, fast and slow, for video question answering. In *IJCNN*, 2020.

[21] Gary F Marcus. *The algebraic mind: Integrating connectionism and cognitive science*. MIT press, 2018.

[22] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.

[23] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *CVPR*, pages 5533–5541, 2017.

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[25] Gursimran Singh, L. Sigal, and J. Little. Spatio-temporal relational reasoning for video question answering. In *BMVC*, 2019.

[26] Xiaomeng Song, Yucheng Shi, Xin Chen, and Yahong Han. Explore multi-step reasoning in video question answering. In *ACM MM*, 2018.

[27] Elizabeth S Spelke and Katherine D Kinzler. Core knowledge. *Developmental science*, 10(1):89–96, 2007.

[28] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *CVPR*, pages 1960–1968, 2019.

[29] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*. IEEE, 2017.

[30] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, 2017.

[31] J. Xu, T. Mei, Ting Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. *CVPR*, pages 5288–5296, 2016.

[32] Yunan Ye, Zhou Zhao, Yimeng Li, Long Chen, Jun Xiao, and Yueting Zhuang. Video question answering via attribute-augmented attention network learning. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 829–832. ACM, 2017.

[33] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. Clevrer: Collision events for video representation and reasoning, 2019.

[34] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. Uncovering the temporal context for video question answering. *International Journal of Computer Vision*, 124(3):409–421, 2017.

[35] Bohan Zhuang, Lingqiao Liu, Chunhua Shen, and Ian Reid. Towards context-aware interaction recognition for visual relationship detection. In *ICCV*, pages 589–598, 2017.