This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



# Relation Regularized Scene Graph Generation

Yuyu Guo, Lianli Gao, *Member, IEEE*, Jingkuan Song, *Senior Member, IEEE*, Peng Wang, Nicu Sebe, *Senior Member, IEEE*, Heng Tao Shen, *Senior Member, IEEE*, and Xuelong Li, *Fellow, IEEE*

***Abstract*—Scene graph generation (SGG) is built on top of detected objects to predict object pairwise visual relations for describing the image content abstraction. Existing works have revealed that if the links between objects are given as prior knowledge, the performance of SGG is significantly improved. Inspired by this observation, in this article, we propose a relation regularized network (R2-Net), which can predict whether there is a relationship between two objects and encode this relation into object feature refinement and better SGG. Specifically, we first construct an affinity matrix among detected objects to represent the probability of a relationship between two objects. Graph convolution networks (GCNs) over this relation affinity matrix are then used as object encoders, producing relation-regularized representations of objects. With these relation-regularized features, our R2-Net can effectively refine object labels and generate scene graphs. Extensive experiments are conducted on the visual genome dataset for three SGG tasks (i.e., predicate classification, scene graph classification, and scene graph detection), demonstrating the effectiveness of our proposed method. Ablation studies also verify the key roles of our proposed components in performance improvement.**

***Index Terms*—Graph convolution networks (GCNs), scene graph generation (SGG), visual relationship.**

Manuscript received October 17, 2019; revised May 13, 2020 and September 17, 2020; accepted January 2, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0102200; in part by the National Natural Science Foundation of China under Grant 61772116, Grant 61872064, Grant 62020106008, and Grant 61871470; in part by the Sichuan Science and Technology Program under Grant 2019JDTD0005; in part by the Open Project of Zhejiang Lab under Grant 2019KD0AB05; and in part by the Open Project of Key Laboratory of Artificial Intelligence, Ministry of Education under Grant AI2019005. This article was recommended by Associate Editor D. Goldgof. *(Corresponding author: Lianli Gao.)*

Yuyu Guo, Lianli Gao, Jingkuan Song, and Heng Tao Shen are with the Future Media Center, University of Electronic Science and Technology of China, Chengdu 611731, China, and also with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: yuyuguo1994@gmail.com; lianli.gao@uestc.edu.cn; jingkuan.song@gmail.com; shenhengtao@hotmail.com).

Peng Wang is with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW 2170, Australia (e-mail: pengw@uow.edu.au).

Nicu Sebe is with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy (e-mail: niculae.sebe@unitn.it).

Xuelong Li is with School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: xuelong_li@nwpu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2021.3052522.

Digital Object Identifier 10.1109/TCYB.2021.3052522

## I. Introduction

IN PRACTICE, providing only object labels and detecting object bounding boxes [1]–[5] may not produce satisfactory semantic information for downstream tasks, such as visual content retrieval [6]–[9]; visual question answering [10], [11]; and visual captioning [12], [13]. For instance, in Fig. 1, generated object labels and bounding boxes (e.g., dog, woman, frisbee, and hair) cannot provide answers to the following question: what are the two dogs and the woman doing? As a result, scene graph generation (SGG) has been proposed and studied by Krishna *et al.* [14], who also collected a dataset consisting of images with objects and object relations to evaluate the quality of the generated scene graph. In the example in Fig. 1, the bottom part represents the scene graph, precisely and briefly describing the semantic content of the top image. With such a scene graph, we can provide an answer to the aforementioned question.

As shown in Fig. 1, a scene graph consists of nodes (objects) and edges (relationships between objects). Given an image, SGG is built on top of detected objects to predict object pairwise visual relations for describing the image content abstraction [15]–[17]. It is challenging to effectively and accurately generate a scene graph and this has been recently been the subject of intensified research [15]–[19]. Some studies have focused on exploiting linguistic priors [15], visual embedding [18], feature interactions [20], external information (e.g., region captions) [21], or spatial information [19] to boost the performance. Some other studies have tried to extract certain contextual information from objects to enhance the performance [16], [17], [22] by iterative message passing, neural motifs, or graphs to avoid detecting and recognizing individual objects in isolation.

From previous works, we can see that the contextual information is important to SGG. To truly produce accurate scene graphs, it is crucial to devise a model that fully and automatically exploits the global contextual information and relational contextual information. The global contextual information encodes the visual information from all objects and backgrounds/environments. The relational contextual information encodes the graph information among objects. Moreover, the previous experimental results of the two SGG tasks [i.e., predicate classification (PREDCLS) and scene graph classification (SGCLS)] [16], [17], [19] have proved that the quality of the detected object labels significantly influences the performance of the SGG. In other words, improving the quality of object label detection could directly lead to better scene graphs.

2168-2267 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



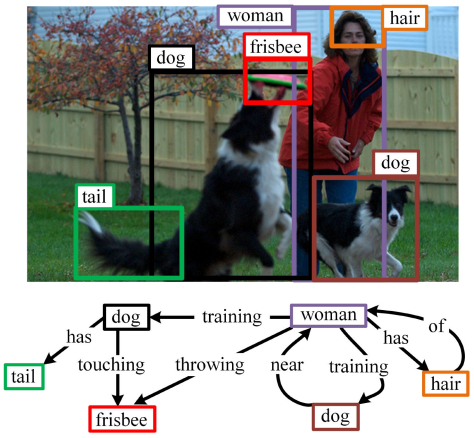



Fig. 1. Ideal SGG method takes an image as input to generate a precise graph for describing the image content abstraction. A scene graph consists of nodes (dog, tail, and so on) and edges (has, touching and so on). It can also be represented by a set of triples (*dog-has-tail*, *dog-touching-frisbee*, and so on).

Therefore, in this article, we address the problem of SGG, where we aim to take advantage of the global context and relational information to produce a relation regularized scene graph from an image. Our contributions are threefold as follows.

1) We propose a novel relation regularized network, namely, R2-Net, which can predict whether there is a relationship between two objects and use this relation as a regularizer to learn relation-embedded features. Therefore, the R2-Net effectively and progressively encodes region features with the global context and relational information to refine object label prediction and SGG.
2) We propose a stacked LSTM-GCN encoder to extract the comprehensive features of objects. Specifically, we stack graph convolution networks (GCNs) on top of bi-LSTMs as an object feature encoder to combine the relation-embedded features and the global features.
3) We verify the effectiveness of our method across three tasks [i.e., PREDCLS, SGCLS, and scene graph detection (SGDET)] on the visual genome dataset. Our ablation study also demonstrates the key roles of our proposed components in performance improvement.

## II. Related Work

In this section, we describe three categories of related works, including: 1) object detection; 2) SGG; and 3) GCNs.

*Object Detection:* Object detection is one of the most fundamental areas in the field of computer vision. Due to the evolution of convolutional neural networks (CNNs) [23], many effective CNN-based methods [4], [5], [24]–[27] have been proposed to deal with this task. First, Girshick *et al.* [24] directly extracted the deep features of warped region proposals with a deep CNN and classified these deep region features with SVMs [28]. However, performing a whole CNN forward pass for each proposal is time consuming. Fast R-CNN [25] extracted the feature map from an entire image and mapped region proposals to regions of interest (RoIs) in the feature map. Then, the RoI pooling layer resized the RoI feature maps to the same size for classification and regression. By sharing the same CNN for region proposals, fast R-CNN saved a lot of computing time. In order to achieve real-time detection, faster R-CNN [5] replaced the time-consuming selective search method with a region proposal network to search the class-agnostic foreground objects. Previous methods with the RoI pooling layer may cause coarse quantization on feature maps. To alleviate this problem, He *et al.* [27] proposed the RoIAlign layer, which preserves precise spatial mappings. Different from the above works, this article focuses on a more complex problem: SGG. To solve this problem, we need not only to detect objects in the image but also to extract the contextual information to predict the relationships among objects.

*Scene Graph Generation:* Object detection cannot adequately represent rich semantic information in images and, therefore, more works [14], [15], [17]–[19], [21], [29], [30] pay attention to SGG (or visual relationship detection) for exploring rich semantic information in images. Since the relationships between objects largely depend on human prior knowledge, Lu *et al.* [15] integrated a visual module and a language module for adequately employing human prior knowledge. Inspired by translating embeddings [31] for modeling multirelational data, Zhang *et al.* [18] proposed the visual translation embedding (VTE) network. In VTE, entities are embedded in a semantic space, and relationships are modeled as a vector translation: subject $-$ object $\approx$ relation. Due to the effectiveness of end-to-end CNNs, Newell and Deng [32] proposed end-to-end convolutional networks that mapped pixels to graphs directly with associative embedding (AE).

All the above works use the local detectors and independently predict relationships between entities. As mentioned in [17], ignoring the surrounding context may lead to ambiguity of model prediction. Therefore, for capturing the surrounding context in images, the authors inferred scene graphs by iteratively refining model predictions with recurrent neural networks. In addition, Zellers *et al.* [16] explored regularly appearing substructures called *motifs* in scene graphs. Inspired by this analysis, the authors proposed a strong baseline. The baseline determines the relationship between two objects through two steps: 1) determining the labels of objects by faster R-CNN [5] and 2) finding the most frequent relationship between the two objects' labels (ignoring the visual information) in the training set. Then, the authors combined the baseline and LSTMs for extracting global context and outperformed the state-of-the-art methods. By analyzing the above works, we find that if the links between objects are given as prior knowledge, the performance of SGG would be significantly improved. Inspired by this observation, we propose an R2-Net, which can predict whether there is a relationship between two objects, and encode this relation with GCNs to refine features and generate robust scene graphs.

*Graph Convolutional Networks:* In order to extract features on graph-structured data, Kipf and Welling [33] proposed GCNs for semisupervised node classification based on spectral graph convolutions. Given a graph, GCNs refine node features

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



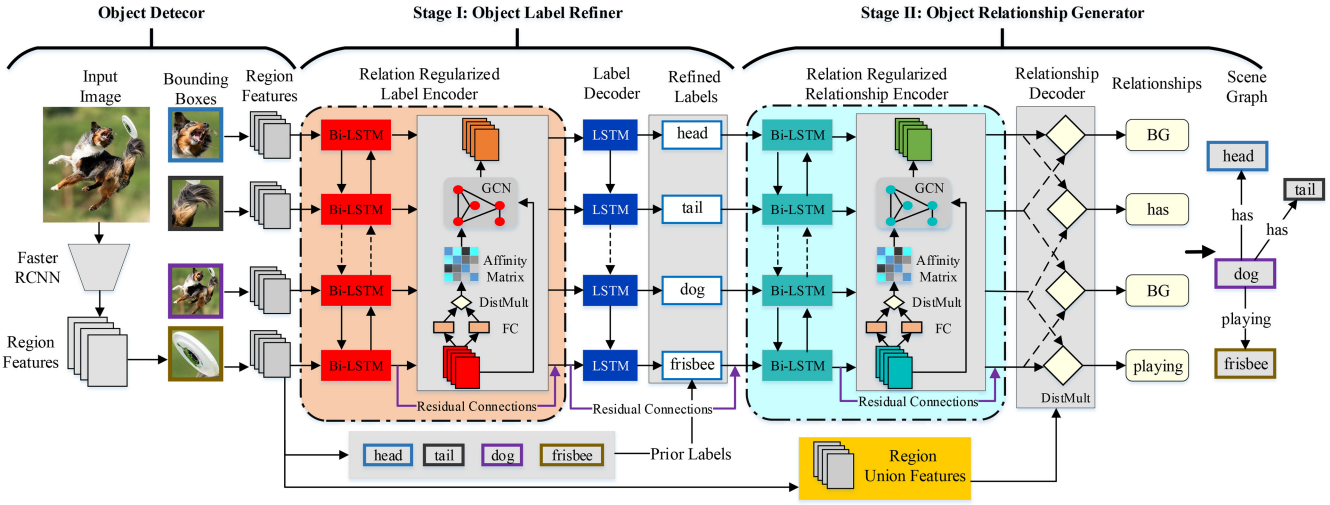



Fig. 2. Framework of R2-Net. After the object detector, our model can be divided into two stages: 1) an object label refiner for refining the prior labels from the object detector and 2) an object relationship generator for generating scene graphs.

TABLE I
TABLE OF MAIN SYMBOLS

| Symbol Name | Dimension | Description |
|---|---|---|
| $D_l$ | $D_l \in \mathbb{Z}^+$ | Total number of object categories. |
| $D_f$ | $D_f \in \mathbb{Z}^+$ | Dimension of features extracted from Faster R-CNN. |
| $D_h$ | $D_h \in \mathbb{Z}^+$ | Dimension of features extracted from Bi-LSTMs. |
| $D_o$ | $D_o \in \mathbb{Z}^+$ | Dimension of output features of the reltion regulrized encoder in the object lable refiner. |
| $D_z$ | $D_z \in \mathbb{Z}^+$ | Dimension of output features of the reltion regulrized encoder in the object relationship generator. |
| $D_r$ | $D_r \in \mathbb{Z}^+$ | Total number of relationship predicate categories. |
| $\mathbf{B} = \{b_1, ..., b_N\}$ | $b_i \in \mathbb{R}^4$ | Bounding boxes generated from Faster R-CNN. |
| $\mathbf{L} = \{l_1, ..., l_N\}$ | $l_i \in \mathbb{R}^{D_l}$ | Label probabilities generated from Faster R-CNN. |
| $\mathbf{F} = \{f_1, ..., f_N\}$ | $f_i \in \mathbb{R}^{D_f}$ | Object features extracted from Faster R-CNN. |
| $\mathbf{U} = \{u_{1,1}, ..., u_{N,N}\}$ | $u_{i,j} \in \mathbb{R}^{D_f}$ | Union features extracted from Faster R-CNN. |
| $\mathbf{H} = \{h_1, h_2, ..., h_N\}$ | $h_i \in \mathbb{R}^{D_h}$ | Global features extracted from Bi-LSTMs. |
| $\mathbf{A}^e = \{a^e_{1,1}, ..., a^e_{N,N}\}$ | $a^e_{i,j} \in \mathbb{R}$ | Affinity matrix of the reltion regulrized encoder in the object label refiner. |
| $\mathbf{O}' = \{o'_1, ..., o'_N\}$ | $o'_i \in \mathbb{R}^{D_o}$ | Output features of the relation regularized encoder in the object label refiner. |
| $\mathbf{A}^r = \{a^r_{1,1}, ..., a^r_{N,N}\}$ | $a^r_{i,j} \in \mathbb{R}$ | Affinity matrix of the reltion regulrized encoder in the object relationship generator. |
| $\mathbf{Z} = \{z_1, \cdots, z_N\}$ | $z_i \in \mathbb{R}^{D_z}$ | Output features of the relaion regulzeried encoder in the object relaionship generator. |
| $\mathbf{R} = \{r_{1,1,1}, ..., r_{D_r,N,N}\}$ | $r_{m,i,j} \in \mathbb{R}$ | Probability that the $i$-th object and the $j$-th object belong to the relationship predicate $m$. |

based on the adjacency matrix and encode graph structures by passing information between adjacent nodes. Due to the effectiveness of GCNs, several works [34]–[36] introduced GCNs into different fields. For instance, Marcheggiani and Titov [35] used GCNs for semantic role labeling (SRL). They stacked GCNs on LSTMs to capture different ranges of information. Schlichtkrull *et al.* [34] proposed relational graph convolutional networks (R-GCNs) for link prediction and entity classification on knowledge graphs. In addition to the graph structure data, GCNs were also applied to computer vision tasks. In order to predict human–object interactions (HOIs) in images, Qi *et al.* [36] expressed HOI structures as graphs. The information between instances can be effectively captured by GCNs. Different from these works, we focus on SGG. Since the task of SGG does not give the affinity matrix in the test phase, we first construct the affinity matrix from the image. Next, GCNs are used to encode the instance features with the affinity graph. In this way, our model can generate robust scene graphs with the graph level features.

## III. RELATION REGULARIZED MODEL

In this article, we propose a relation regularized model for the SGG. The overview of our proposed model is depicted in Fig. 2. The framework consists of three components: 1) object detection (i.e., bounding box detection and prior object label detection); 2) relation regularized label refiner; and 3) relation regularized relationship generation. In the following sections, we first introduce the definition of our problem and then describe the details of the model from inputs to outputs (object detector, object label refiner, object relationship generator, and loss functions). Since this section contains many symbols, we show the dimensions and descriptions of the main symbols in Table I.

### A. Problem Definition

We define the image SGG problem as follows: given an image $\mathbf{I}$, we want to generate a scene graph $\mathbf{G}$ to describe its content abstraction. Following previous works [6], [16], we determine to progressively decompose the SGG into a series of

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



continued actions: bounding boxes detection **B**, bounding box label detection **L**, and label relation detection **R**. Therefore, **G** is defined as $\mathbf{G} = \{\mathbf{B}, \mathbf{L}, \mathbf{R}\}$. The probability of **G** is decomposed by a multiplication rule

$$\begin{aligned} P(\mathbf{G}|\mathbf{I}) &= P(\mathbf{B}, \mathbf{L}, \mathbf{R}|\mathbf{I}) \\ &= P(\mathbf{B}|\mathbf{I})P(\mathbf{L}|\mathbf{B}, \mathbf{I})P(\mathbf{R}|\mathbf{L}, \mathbf{B}, \mathbf{I}). \end{aligned} \tag{1}$$

### B. Object Detector

The object detector is utilized to detect instances in images. The inputs of the object detector are images, and the outputs are bounding boxes, categories, and features of instances. Faster R-CNN [5] has achieved great success in image object detection, and it has been widely adopted to support image SGG [16]–[18], [22]. In this article, we adopt faster R-CNN to generate a set of bounding boxes $\mathbf{B} = \{b_1, \ldots, b_N\}$, where $b_i \in \mathbb{R}^4$. $N$ is the total number of detected bounding boxes from the input image. $i$ is an index ranging from 1 to $N$. With $N$ boxes, we can further obtain as follows.

1) A set of label probabilities $\mathbf{L} = \{l_1, \ldots, l_N\}$, where $l_i \in \mathbb{R}^{D_l}$ and $D_l$ is the total number of labels in a dataset.
2) A set of object feature vectors $\mathbf{F} = \{f_1, \ldots, f_N\}$, where $f_i \in \mathbb{R}^{D_f}$ and $D_f$ is the feature dimension.
3) A set of feature vectors of union boxes $\mathbf{U} = \{u_{1,1}, \ldots, u_{N,N}\}$, where $u_{i,j} \in \mathbb{R}^{D_f}$. Each union box is the smallest rectangle containing two bounding boxes.

### C. Stage I: Object Label Refiner

To efficiently and effectively evaluate an SGG model, previous works [16]–[18], [22] have designed two experiments: 1) SGCLS and 2) PREDCLS (see details in Section IV-C). All existing experimental results [16]–[18], [22] have shown that the scores (recall@20, 50, 100) of PREDCLS are significantly higher than those of SGCLS by approximately 30%. Both SGCLS and PREDCLS take ground-truth boxes as inputs, but PREDCLS adopts the ground-truth labels, while SGCLS takes the predicted object label. These results [16] prove the importance of accurate object labels for SGG. In other words, how to improve the prediction accuracy of object labels is the key to solving the SGG problem.

Therefore, in the second step, we aim to improve SGG by refining the object labels generated by the faster R-CNN network. The inputs of the object label refiner are instance features, bounding boxes, and categories, and the outputs are refined object categories. When generating object labels, the faster R-CNN neither considers the global context [16] nor object relations. Thus, we propose a relation regularized module for label refinement with a stack of bi-LSTM [37]–[39] to capture the global context and a relation-based graph convolution layer [33]–[35] to make full use of object relationships.

*Relation Regularized Label Encoder:* In our work, we use deep bidirectional LSTMs to explore the global context. However, directly using deep LSTMs may exist the problems of training difficulty and slow convergence. In order to alleviate such issues, the highway LSTM [40] was designed by connecting memory units of adjacent LSTM layers, and this structure [38], [40] has been used predominantly to achieve state-of-the-art results for semantic labeling, language modeling, etc. Therefore, we use stacked bi-LSTMs with highway connections to encode object features **F** to alleviate the problems caused by deep LSTMs

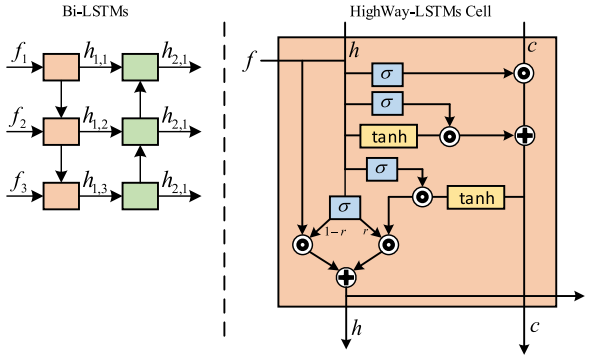



Fig. 3. Left figure shows the data stream of bi-LSTMs. The right figure shows the cell of bi-LSTMs with the highway connection. The operator $\circ$ represents the Hadamard product (also known as the elementwise product). The function $\sigma$ represents the sigmoid function. For clarity, we omit the subscript of the symbol in the right figure.

$$\begin{aligned} i_{k,t} &= \sigma\left(W_i^k\left[h_{k,t+\delta_k}, x_{k,t}\right] + b_i^k\right) \\ o_{k,t} &= \sigma\left(W_o^k\left[h_{k,t+\delta_k}, x_{k,t}\right] + b_o^k\right) \\ f_{k,t} &= \sigma\left(W_f^k\left[h_{k,t+\delta_k}, x_{k,t}\right] + b_f^k\right) \\ r_{k,t} &= \sigma\left(W_r^k\left[h_{k,t+\delta_k}, x_{k,t}\right] + b_r^k\right) \\ g_{k,t} &= \tanh\left(W_g^k\left[h_{k,t+\delta_k}, x_{k,t}\right] + b_g^k\right) \\ c_{k,t} &= f_{k,t} \circ c_{k,t+\delta_k} + i_{k,t} \circ g_{k,t} \\ h'_{k,t} &= o_{k,t} \circ \tanh(c_{k,t}) \\ h_{k,t} &= r_{k,t} \circ h'_{k,t} + \left(1 - r_{k,t}\right) \circ W_h^k x_{k,t} \end{aligned} \tag{2}$$

where $x_{k,t}$ is the input to the $k$th LSTM layer at time step $t$, $W_*^*$, and $b_*^*$ are parameters, and $\delta_k$ indicates the direction of the $k$th LSTM layer. The operator $\circ$ represents the Hadamard product (also known as the elementwise product). The function $\sigma$ represents the sigmoid function. Following He *et al.* [38], we set the inputs $x_{k,t}$ and $\delta_k$ of each LSTM layer as follows:

$$x_{k,t} = \begin{cases} f_t, & k = 1 \\ h_{k-1,t}, & k > 1 \end{cases} \tag{3}$$

$$\delta_k = \begin{cases} 1, & k \bmod 2 = 0 \\ -1, & k \bmod 2 = 1 \end{cases} \tag{4}$$

where $f_t$ is the $t$th bounding box feature of **F** extracted from the faster R-CNN as mentioned in Section III-B. The outputs of the highway LSTMs are denoted as $\mathbf{H} = \{h_1, h_2, \ldots, h_N\}$. The data stream of bi-LSTMs with highway connection is shown as Fig. 3.

GCNs [33]–[35] produce an optimized node-level output, which encodes both original node features and the associations between data nodes. By using GCNs, our model can integrate the information of related objects to boost the performance of the label prediction.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



With faster R-CNN and stacked bi-LSTMs, we can obtain a set of global features $\mathbf{H} = \{h_1, h_2, \ldots, h_N\}$ corresponding to previously detected $N$ objects. By considering each object feature as a node, we take the global features $\mathbf{H}$ to estimate an adjacency matrix $\mathbf{A}^e = \{a^e_{1,1}, \ldots, a^e_{N,N}\}$, where $a^e_{i,j} \in \mathbb{R}^1$ represents whether a relationship exists from object $i$ to object $j$.

In the process of SGG, an object label may act as a subject or an object in a scene graph. Thus, we first adopt two fully connected layers to map $h_i$ into a subject space domain and an object space domain, respectively

$$\begin{aligned} h^s_i &= FC^s_h(h_i) \\ h^o_i &= FC^o_h(h_i). \end{aligned} \tag{5}$$

The output dimensions of the two fully connected layers are $D_f$. Next, we apply a simple and effective scoring function, DistMult [41], to compute affinity matrix scores

$$a^e_{i,j} = \sigma\Big(\big(h^s_i \circ u_{i,j}\big)^T W^a \Big(h^o_j \circ u_{i,j}\Big) + b^a_{i,j}\Big) \tag{6}$$

where $W^a \in \mathbb{R}^{D_f \times D_f}$ is a diagonal parameter matrix that the model needs to learn. $b^a_{i,j} \in \mathbb{R}^1$ is a bias specific to the subject $i$ and object $j$ labels. Following [16], we initialize the bias with the frequency of the training set. $\sigma$ is an activation function mapping the score ranging from 0 to 1. Besides, giving two objects it is difficult to determine the information flow direction from the object to the subject or vice versa. Therefore, the adjacency matrix is adjusted to form a symmetric matrix $\mathbf{A}^s$ to solve this issue

$$a^s_{i,j} = \begin{cases} a^e_{i,j}, & \text{if } a^e_{i,j} \geq a^e_{j,i} \\ a^e_{j,i}, & \text{if } a^e_{i,j} < a^e_{j,i} \\ 1, & \text{if } i = j. \end{cases} \tag{7}$$

With the generated symmetric matrix $\mathbf{A}^s$, we integrate bi-LSTMS with GCNs to obtain relational features $\mathbf{O} = \{o_1, \ldots, o_N\}$

$$\mathbf{O} = ReLU\Big(\mathbf{D}^s \mathbf{A}^s \mathbf{H} W^G\Big) \tag{8}$$

where $W^G$ is a parameter matrix. $\mathbf{D}^s = \{d^s_{1,1}, \ldots, d^s_{N,N}\}$ is a diagonal matrix

$$d^s_{i,j} = \begin{cases} \frac{1}{\sum_{k=1}^{N} a^s_{i,k}}, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}. \tag{9}$$

Next, we concatenate the global context $h_i$ and the relational feature $o_i$ to form the final output $o'_i$

$$o'_i = [o_i, h_i]. \tag{10}$$

For simplicity, we define the whole relation regularized encoding process as

$$\big\{\mathbf{A}^e, \mathbf{O}'\big\} = \text{R2\_Encoder}(\mathbf{F}|W^o) \tag{11}$$

where $W^o$ is the parameter in the relation regularized module.

*Label Decoder:* Finally, we use an LSTM layer with highway gate to decode $\mathbf{O}'$. After refining the previous generated initial object labels $\mathbf{L}$, we obtain the refined object label $\mathbf{L}^d$

$$\begin{aligned} q^d_i &= \text{LSTM}\Big(\Big[l^d_{i-1}, q^d_{i-1}, o'_i\Big]\Big) \\ l^d_i &= \text{argmax}\Big(W q^d_i + l_i\Big) \end{aligned} \tag{12}$$

where $l_i$ is the prior label distribution from faster R-CNN as mentioned in Section III-B. $q^d_i$ is the hidden state of LSTM. $l^d_i$ is the $i$th refined object labels $\mathbf{L}^d$. In addition, we set $\langle BOS \rangle$ as the start signal for the decoding process.

### D. Stage II: Object Relationship Generator

The object relationship generator predicts the relationship predicate $\mathbf{R}$ between instances with instance features $\mathbf{O}'$ from (11) and refined labels $\mathbf{L}$ from (12).

*Relation Regularized Relationship Encoder:* With faster R-CNN and our proposed relation regularized label generation module, we can transfer an image into a set of optimized instance features $\mathbf{O}'$ and refined labels $\mathbf{L}^d$. Next, we apply our proposed relation regularized encoder to facilitate the relationship prediction process

$$\big\{\mathbf{A}^r, \mathbf{Z}\big\} = \text{R2\_Encoder}\Big(\Big[\mathbf{O}', W^L \mathbf{L}^d\Big]\Big| W^z\Big) \tag{13}$$

where $W^L$ is the embedding parameter and is initialized by Glove [42], and $\mathbf{Z} = \{z_1, \ldots, z_N\}$ is the output feature of R2-Encoder in the relationship generator phase. $\mathbf{Z}$ is similar to $\mathbf{O}'$ in (11) and contains the global and relational features in the relationship generator.

*Relationship Decoder:* Next, we obtain the object relationship by first mapping $z_i$ into two feature space domains: 1) subject domain and 2) object domain, $z^s_i$ and $z^o_i$, respectively

$$\begin{aligned} z^s_i &= FC^s_h(z_i) \\ z^o_i &= FC^o_h(z_i). \end{aligned} \tag{14}$$

We then employ the DisMult score function to compute the object relations

$$r'_{m,i,j} = \big(z^s_i \circ u_{i,j}\big)^T W^r_m \Big(z^o_j \circ u_{i,j}\Big) + b^r_{m,i,j} \tag{15}$$

where $r'_{m,i,j}$ is the probability that the object $i$ and object $j$ belong to the relationship $m$. $W^r_m$ is a diagonal parameter matrix that needs to be learned. $b^r_{m,i,j}$ is the frequency bias mentioned in [16]. Finally, we use a softmax function to obtain the final relation score ranging from 0 to 1

$$r_{m,i,j} = \frac{e^{r'_{m,i,j}}}{\sum_{m=1}^{D_r} e^{r'_{m,i,j}}} \tag{16}$$

where $D_r$ is the number of relationship categories in the dataset. Now we can get the final relationships $\mathbf{R} = \{r_{1,1,1}, \ldots, r_{D_r,N,N}\}$.

### E. Loss Functions

In this section, we first describe two objective functions. The first one is the label prediction loss $\mathcal{L}_1$ for refining the object labels, while the second loss is relation regularized loss

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(R2-loss $\mathcal{L}_2$) for learning the first adjacency matrix

$$\mathcal{L}_1 = \text{Cross\_Entropy}\left(\mathbf{L}^d, \mathbf{L}^g\right)$$
$$\mathcal{L}_2 = \text{Cross\_Entropy}\left(\mathbf{A}^e, \mathbf{A}^g\right) \tag{17}$$

where $\mathbf{L}^d$ is the output from (12), and $\mathbf{L}^g$ is the ground-truth object labels. $\mathbf{A}^e$ is obtained by (6). $\mathbf{A}^g$ is the ground-truth adjacency matrix, which is used to indicate whether there is a relationship between two entities, that is, 0 or 1.

To learn the parameters for generating relations between objects, we describe two object functions to control our model

$$\mathcal{L}_3 = \text{Cross\_Entropy}\left(\mathbf{A}^r, \mathbf{A}^g\right)$$
$$\mathcal{L}_4 = \text{Cross\_Entropy}\left(\mathbf{R}, \mathbf{R}^g\right) \tag{18}$$

where $\mathcal{L}_3$ is another relation regularized loss at the relationship generation phase, and $\mathcal{L}_4$ is the object relationship loss (R2-loss). The proposed final objective function of our proposed a relation regularized model is defined as the sum of $\mathcal{L}_1$, $\mathcal{L}_2$, $\mathcal{L}_3$, and $\mathcal{L}_4$.

## IV. Experiments

In this section, we evaluate our model on the cleaned visual genome dataset [14]. Some experiments are conducted to test the role of the major components and to compare our model with the previous methods.

### A. Experimental Setting: Dataset

Krishna *et al.* [14] collected and officially released a knowledge base to connect structured images concept to languages, namely, visual genome dataset. They have collected more than 100k images. On average, each image contains 38 objects and 22 pairs of object relations. This is an ideal candidate dataset for evaluating SGG models. However, some annotations are ambiguous and may lead to predicting errors. Many cleaning approaches have been proposed, such as [17], [21], and [29]. In particular, Xu *et al.* [17] proposed a cleaning strategy to remove ambiguous annotations. This strategy has been widely adopted by previous SGG methods, such as [15]–[17], [22], and [32]. After cleaning, each image, on average, contains approximately 12 objects and six pairs of relationships. In total, the cleaned visual genome dataset contains 150 object categories and 50 object relation classes. Moreover, we follow Xu *et al.* [17] to divide the cleaned dataset into two subsets: 1) 70% training and 2) 30% testing. Next, we follow [15]–[17], [22], [32] to select 5k images from the training dataset as the validation set.

### B. Experimental Setting: Implementation Details

For object detection step, we follow the previous works [16], [17] to adopt the faster R-CNN model as the object detector to generate a set of bounding boxes and their corresponding features and labels. Specifically, previous works [16], [17], [22], [32], and [43] adopted the VGG16 network as the backbone of pretrained faster R-CNN. For a fair comparison, we also adopt the VGG16-based faster R-CNN network, which is pretrained on the visual genome object dataset by Zellers *et al.* [16]. For the VGG16-based faster R-CNN network, we obtain the region features from the output feature map of the second fully connected layer.

By using ResNet101 to extract deeper features, previous work [5] has proven that ResNet101-based faster R-CNN can perform better than VGG16-based faster R-CNN on the object detection task. Better region features and more accurate object labels may be conducive to object relation generation. Therefore, we first take the ResNet101 [3] as the backbone of the faster R-CNN and then train the ResNet101-based faster R-CNN on the training set of the cleaned visual genome dataset. Specifically, the detector is trained on a Titan Xp (12 GB) with the SGD optimizer. The learning rate is set as $1 \times 10^{-3}$ and the batch size is set as 6. We compare the VGG16-based faster R-CNN model with the ResNet101-based one on the cleaned test dataset in ablation studies. In addition, we take the *pool*_5 layer of ResNet101 as the output feature map for extracting region features.

For the second step, label refinement, our relation regularized encoder requires the input regions to be sorted in an order. Therefore, we sort the object features from left to right as our encoder's inputs. A previous study [16] has proven that the order of the bi-LSTMs inputs has little effect to extract global context-based region features. For the first relation regularized encoder, we set the number of bi-LSTM layers as 2. For the second relation regularized encoder of relation generation, we set the number of bi-LSTM layers as 4.

We first train the detector (faster R-CNN) on the visual genome dataset as mentioned above, and then freeze the parameters of faster R-CNN when processing the SGG. When training our model on the SGCLS and the PREDCLS, we follow previous works [17], [32] to use the ground-truth bounding boxes for refining object labels and generating object relations. In order to learn two object relation matrices $A^e$ and $A^r$, we sample twice as many positive samples as negative samples. For relation $\mathbf{R}$, we utilize the same number of positive and negative samples. Moreover, we choose SGD with momentum [44] as our optimizer. The learning rate and batch size are set to $2 \times 10^{-2}$ and 24, respectively. When training our model on the SGDET, we fine-tune our model following the strategy mentioned in [16] for fairness. We sample 256 RoIs in each image. After the label decoder in the object label refiner, we use the per-category nonmaximum suppression (NMS) to filter redundant objects.

Our codes are implemented in python. We use Pytorch to build our model. All experiments are conducted on a Ubuntu server with 2 Titan Xps (12 GB), 4 Intel Xeon E5-2650 CPUs and 256-GB RAM.

### C. Experimental Setting: Evaluation Strategy

Following previous works [17], [22], [32], we evaluate our model with three experimental setups: 1) the PREDCLS allows models to take the ground-truth bounding boxes and the ground-truth object labels as inputs; 2) the SGCLS allows models to take the ground-truth bounding boxes as inputs; and 3) the SGDET, which requires a model to take an image as inputs and then predict object bounding boxes, object labels,

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



TABLE II
DIFFERENCES AMONG THREE TASKS: PREDCLS, SGCLS, AND SGDET. CHECK MARKS INDICATE WHAT INFORMATION A TASK NEEDS TO PREDICT

| Tasks | Required Prediction | | |
|---|---|---|---|
| | Relationship Predicate | Object Label | Object Bounding Box |
| PREDCLS | ✓ | | |
| SGCLS | ✓ | ✓ | |
| SGDET | ✓ | ✓ | ✓ |

TABLE III
ROLE OF RELATION REGULARIZED LABEL REFINER. THE RESULTS ARE OBTAINED ON THE VALIDATION DATASET

| R2 Variants | SGCLS | | | |
|---|---|---|---|---|
| | R@20 | R@50 | R@100 | obj acc |
| w/o refiner | 33.3 | 39.1 | 41.1 | 71.1 |
| w/o prior labels | 38.7 | 42.1 | 42.9 | 71.5 |
| All | 39.9 | 42.6 | 43.3 | 72.3 |

and object relations. Table II shows what information a task needs to predict. Since the PREDCLS task allows models to take the ground-truth bounding boxes and the ground-truth object labels as inputs, it only requires models to predict relational predicates. However, the SGCLS task requires models to predict relationship predicates and object labels, and the SGDET task requires models to predict relationship predicates, object labels and object bounding boxes. In other words, SGDET is the most difficult one, while PREDCLS is the easiest one. Moreover, in order to provide a more consistent comparison, we also report the corresponding results of the three setups without scene graph constraints.

The object relations in the visual genome dataset are sparse, thus using the mean average precision (mAP) as the evaluation metric would falsely punish positive predictions on unannotated relations. As a result, we follow previous works [15]–[17], [32] to evaluate our model with recall@K (i.e., R@K). Specifically, R@K describes the proportion of ground-truth triplets (i.e., obj1-relation-obj2) in the top $K$ predicted triplets. The $K$ is set as 20, 50, and 100. Besides, for the *SGDET* task, if the object has at least 0.5 IoU overlapping with the ground-truth box, it is considered as correctly detected.

### D. Results and Analysis: Speed of R2-Net

Regarding the speed of our model, in the training phase, each epoch (about 57k images) takes 40 min for the SGCLS task and 210 min for SGDET. In the testing phase, our model takes about 0.12 s to parse a single image for SGCLS and PREDCLS, and 0.33 s for SGDET. These results are obtained under the experimental environment mentioned in Section IV-B.

### E. Results and Analysis: Ablation Study

In order to deeply analyze the proposed approach and demonstrate its effectiveness, we present an extensive ablation study on the visual genome dataset by considering different variants of the proposed R2-Net to evaluate its major components. In this section, the experimental results are obtained from the validation dataset and we choose ResNet101-based faster R-CNN network as our object detector.

*Effect of Object Label Refiner:* We consider three R2-Net variants: 1) *w/o refiner* where R2-Net performs the SGG with only two components. The object detector is followed by an object relation generator; 2) *w/o prior labels* where the object label refiner without the prior labels mentioned in (12); and 3) *All* where R2-Net performs the task with all three components: a) object detector; b) object label refiner; and c) object relation generator. We carry out the experiments, as shown in Table III. It can be observed that *all* significantly performs better than *w/o refiner* on the *SGCLS* task with an increase of 6.6% on R@20, 3.5% on R@50, and 2.2% on R@100, while slightly outperforms the label detection by 1.2% in terms of accuracy. The experimental results prove that our proposed label refiner is able to extract more representative features for both object label classification and SGG. Compared with *w/o prior labels*, using prior labels can improve 1.2% on SGCLS (R@20) and 0.8% on object accuracy. Therefore, we use prior labels in the object label refiner to make the model easier to learn.

*Effect of Relation Regularized Encoder:* Our R2-Net consists of two relation regularized encoders. The first one is proposed for supporting the object label refiner, while the second one is designed for facilitating the object relationship generation. Both encoders consist of two components: 1) highway-based bi-LSTMS for capturing the global context and 2) object relation-based graph convolutional layers for learning object relations. In order to deeply evaluate the effect of bi-LSTMs and relation regularized GCNs of the two encoders, we design a set of R2-Net variants by removing the bi-LSTMs of the first encoder (w/o bi-LSTM1), removing the bi-LSTMs of the second encoder (w/o bi-LSTM2), removing the GCNs of the first encoder (w/o GCN1), removing the GCNs of the second encoder (w/o GCN2), removing all GCNs of the two encoders (w/o GCNs), removing all bi-LSTMs of the two encoders (w/o bi-LSTMs), and the full R2-Net (All).

The experimental results are shown in Table IV and we have the following observations.

1) Compared with All, w/o bi-LSTM1 decreases the R2-Net performance more by 0.8% R@20, 0.7% R@50 and 0.6% R@100. This demonstrates the effectiveness of the proposed bi-LSTM1 for encoding global context.
2) Removing either GCNs (GCN1 or GCN2) could lead to a drop in both tasks, SGG and label prediction. However, without GCN1, the performance scores drop dramatically.
3) Compared with other R2-Net variants, w/o GCNs obtains the lowest scores for R@20 with 38.3%, which is 1.6% lower than R2-Net. This demonstrates the superiority of the proposed GCNs for relation graph generation.
4) Compared with w/o R2-loss [mentioned in (17) and (18)], using the relation regularized loss can bring additional supervision information to the model and improve the robustness of the model.
5) Compared with all variants, R2-Net (All) performs best on all tasks.
6) Another fact is that w/o GCN2 and w/o bi-LSTM2 have little effects on the SGCLS task. This shows that the

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



TABLE IV
ROLE OF BI-LSTMS AND GCNS IN R2-NET.THE RESULTS ARE OBTAINED ON THE VALIDATION DATASET

| R2 Variants | SGCLS | | | |
|---|---|---|---|---|
| | R@20 | R@50 | R@100 | obj acc |
| w/o Bi-LSTM1 | 39.1 | 41.9 | 42.7 | 71.8 |
| w/o Bi-LSTM2 | 39.7 | 42.3 | 43.1 | 72.2 |
| w/o GCN1 | 39.0 | 41.8 | 42.5 | 71.8 |
| w/o GCN2 | 39.4 | 42.4 | 43.2 | 72.1 |
| w/o GCNs | 38.3 | 42.0 | 43.0 | 72.2 |
| w/o Bi-LSTMs | 39.2 | 42.0 | 42.8 | 71.9 |
| w/o R2-Loss | 38.4 | 42.1 | 43.0 | 71.9 |
| All | 39.9 | 42.6 | 43.3 | 72.3 |

TABLE V
ROLE OF TWO DEEP LEARNING FEATURES: VGG16 AND RESNET101. THE RESULTS ARE OBTAINED ON THE TESTING DATASET

| Model | SGCLS | | | PREDCLS | | |
|---|---|---|---|---|---|---|
| | R@20 | R@50 | R@100 | R@20 | R@50 | R@100 |
| Motifnet(VGG) [16] | 32.9 | 35.8 | 36.5 | 58.5 | 65.2 | 67.1 |
| Motifnet(ResNet) | 33.1 | 36.0 | 36.7 | 58.5 | 65.0 | 66.8 |
| R2-Net(VGG) | 33.5 | 36.5 | 37.3 | **59.2** | **65.9** | **67.8** |
| R2-Net(ResNet) | **34.5** | **37.5** | **38.3** | 59.0 | 65.5 | 67.3 |

prediction of labels is very important on the SGCLS task because GCN2 and bi-LSTM2 are at the relationship generation phase. This further proves that each component is helpful and contributes to the final object detection and SGG.

*Effect of CNN Backbones:* As mentioned in Section IV-B, the different backbones of the object detector can make the model produce different performances, so we compare the different backbones (VGG16 and ResNet101) in this section.

We first compare the performances of these two backbones on the object detection task. The ResNet101-based faster R-CNN performs better than the VGG16-based model (22.8 mAP versus 20.0 mAP at 0.5 IoU) on the visual genome dataset. This result fully demonstrates the advantages of ResNet.

We further evaluate the effect of CNN backbones of the detector on the task of *SGCLS* and *PREDCLS*. Here, we choose to run the best previous method Motifnet and our proposed R2-Net to obtain the results on both tasks. The experimental results are shown in Table V. Specifically, the experimental results demonstrate that on the *SGCLS* task, the ResNet101-based model is obviously better than the corresponding VGG16-based model. Interestingly on the *PREDCLS* task, the VGG16-based models achieve higher scores than the corresponding ResNet101-based ones. In addition, with the same evaluation metrics and the same backbones, the performance score obtained on the *PREDCLS* task is around twice higher than the score reached from the *SGCLS*. From the experimental results, we can conclude that SGG is largely depending on the accuracy of the object label prediction, while the label prediction accuracy is mostly relying on the region features. The more representative the region feature is the more accurate the object label prediction is. Moreover, from the experimental results, we can see that higher level features have only a slight effect on object label relation generation. More importantly, our R2-models significantly outperform the previous best method Motifnet on both two tasks, reaching the new state of the art.

TABLE VI
PERFORMANCE PER PREDICATE (TOP TEN ON LEFT, BOTTOM TEN ON RIGHT)

| Predicate | Recall | Predicate | Recall |
|---|---|---|---|
| wearing | 96.2 | across | 13.7 |
| on | 93.4 | painted on | 11.6 |
| riding | 92.8 | against | 9.7 |
| has | 91.9 | between | 9.5 |
| of | 91.5 | mounted on | 8.9 |
| wears | 86.9 | growing on | 3.4 |
| holding | 86.1 | playing | 1.0 |
| in | 82.6 | from | 0.0 |
| walking on | 81.9 | says | 0.0 |
| sitting on | 78.9 | flying in | 0.0 |

*Which of the Predicates Perform Better/Worse?* We investigate the performance of each predicate on the PREDCLS task (R@100) without graph constraints. In Table VI, the top ten predicates with the highest score are shown on the left side and the bottom ten predicates with the lowest score are shown on the right side. By analyzing the dataset, we find that the samples of the top ten categories account for 78.9% of the total dataset, while the samples of the bottom ten categories account for only 1.3% of the total dataset. Therefore, the imbalance of the dataset is the main reason for the huge difference in predicate scores.

### F. Results and Analysis: Comparison With State-of-the-Art Methods

In this section, we compare our proposed R2-Net with several state-of-the-art methods, including visual relation detection (VRD) [15], iterative message passing (IMP) [17], tensorize factorize regularize (TFR) [45], AE [32], graph R-CNN [22], FREQ+OVERLAP [16], and Motifnet [16]. We cannot compare our method with Factorizable Net [19], DR-Net [29], and MSDN [21], because the approaches for data cleaning and splitting are different. Besides, Factorizable Net [19] has proven that the more bounding boxes generated, the better performances of SGG are. However, the more bounding boxes are chosen, the more complex the computation is. Motivated by this, for the task of *SGDET*, we choose the top 64 regions detected by the faster R-CNN, following previous works [16], [17], [45]. To fully evaluate our method, we apply two evaluation strategies: with and without graph constraints. The experimental results are shown in Table VII (with graph constraints) and Table VIII (without graph constraints). From them, we can see that with two experimental settings, our R2-Net performs the best on all three tasks with the supervised learning strategy, which confirms the effectiveness of our proposed method. Compared with the methods using reinforcement learning, our R2-Net (w/o GCN1) still achieves better results on the SGDET task. Because it is not practical to provide the ground truths of object bounding boxes or categories for models in real life, the SGDET task has more practical value than SGCLS or PREDCLS. Therefore, our method is superior to VCTREE and CMAT in some aspects.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



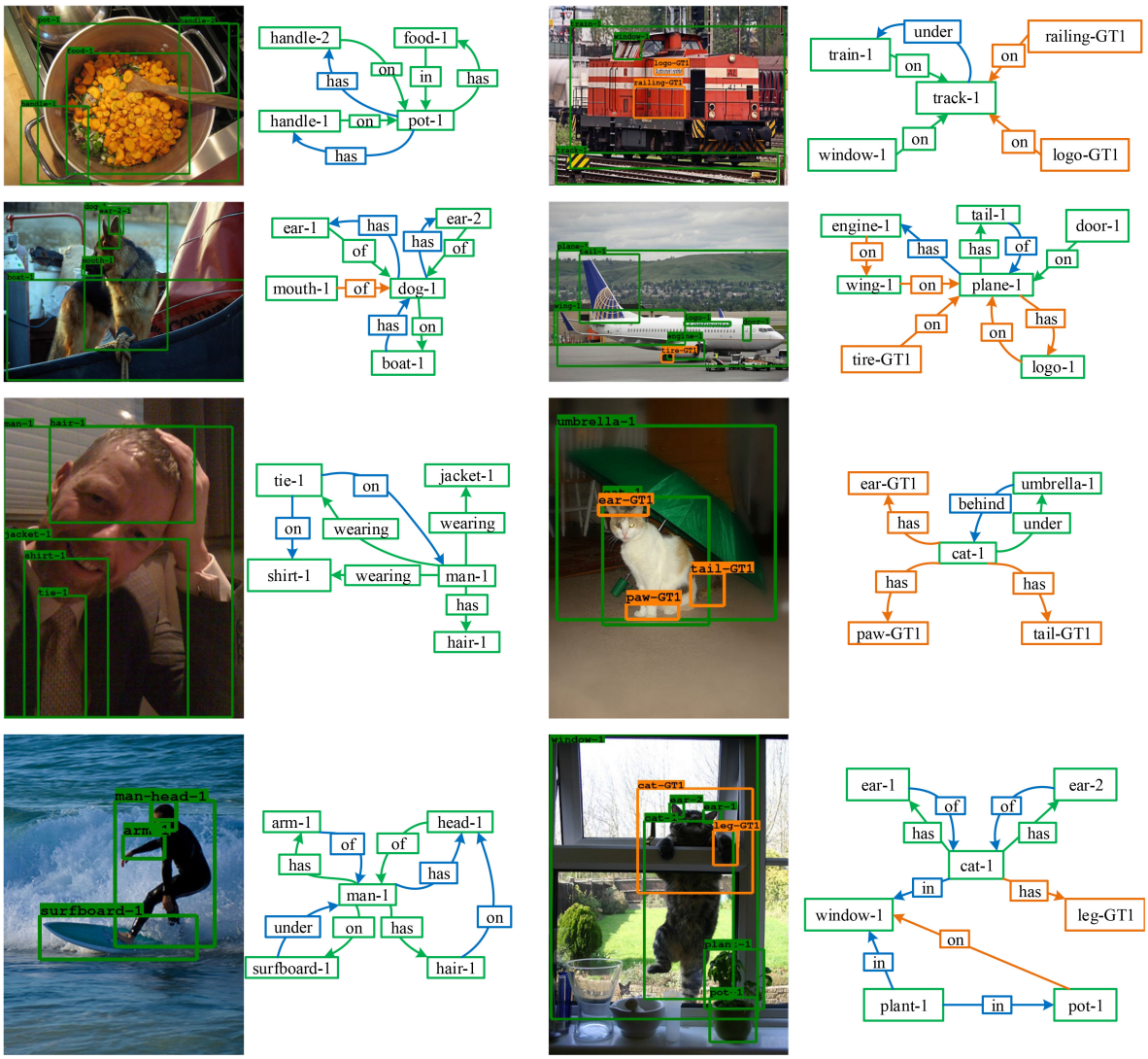



Fig. 4. Some qualitative examples produced by our R2-Net (w/o GCN1) on the *SGDET*. Predicted true positive boxes are marked with green (IOU> 0.5). Orange boxes are ground-truth boxes but not detected. For simplicity, we demonstrate the top 20 object relations or edges at the R@20 setting. Predicted true positives relations are marked with green arrows, false-negative relations are marked with orange arrows, and false-positive relations are marked with blue arrows.

TABLE VII
COMPARISON WITH OTHER METHODS. THE RESULTS ARE OBTAINED ON THE TEST DATASET. THE RESULTS OF IMP+ ARE REPRODUCED IN [16]. THE SUPERVISED LEARNING STRATEGY IS DENOTED AS SL. THE REINFORCEMENT LEARNING STRATEGY IS DENOTED AS RL

| Learning Strategy | Model | SGDET | | | SGCLS | | | PREDCLS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | R@20 | R@50 | R@100 | R@20 | R@50 | R@100 | R@20 | R@50 | R@100 |
| SL | VRD [15] | - | 0.3 | 0.5 | - | 11.8 | 14.1 | - | 27.9 | 35.0 |
| | IMP [17] | - | 3.4 | 4.2 | - | 21.7 | 24.4 | - | 44.8 | 53.0 |
| | IMP+ [17] | 14.6 | 20.7 | 24.5 | 31.7 | 34.6 | 35.4 | 52.7 | 59.3 | 61.3 |
| | TFR [45] | 3.4 | 4.8 | 6.0 | 19.6 | 24.3 | 26.6 | 40.1 | 51.9 | 58.3 |
| | AE [32] | 6.5 | 8.1 | 8.2 | 18.2 | 21.8 | 22.6 | 47.9 | 54.1 | 55.4 |
| | FREQ+OVERLAP [16] | 20.1 | 26.2 | 30.1 | 29.3 | 32.3 | 32.9 | 53.6 | 60.6 | 62.2 |
| | Graph R-CNN [22] | - | 11.4 | 13.7 | - | 29.6 | 31.6 | - | 54.2 | 59.1 |
| | Motifnet [16] | 21.4 | 27.2 | 30.3 | 32.9 | 35.8 | 36.5 | 58.5 | 65.2 | 67.1 |
| | R2-Net (w/o GCN1) | **23.1** | **29.4** | **33.0** | 33.6 | 36.5 | 37.3 | 58.8 | **65.6** | **67.4** |
| | R2-Net | 21.6 | 27.5 | 31.3 | **34.5** | **37.5** | **38.3** | **59.0** | 65.5 | 67.3 |
| SL+RL | VCTREE [46] | 22.0 | 27.9 | 31.3 | 35.2 | 38.1 | 38.8 | 60.1 | 66.4 | 68.1 |
| | CMAT [47] | 22.1 | 27.9 | 31.2 | 35.9 | 39.0 | 39.8 | 60.2 | 66.4 | 68.1 |

Specifically, as shown in Table VII, R2-Net without graph convolutional layers (GCN1) of the first relation regularized encoder achieves the highest scores for the *SGDET* task, reaching 23.1% on R@20, 29.4% on R@50, and 33.0% on R@100, which is significantly higher than the R2-Net. However, for the task of *SGCLS*, the performance result is

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



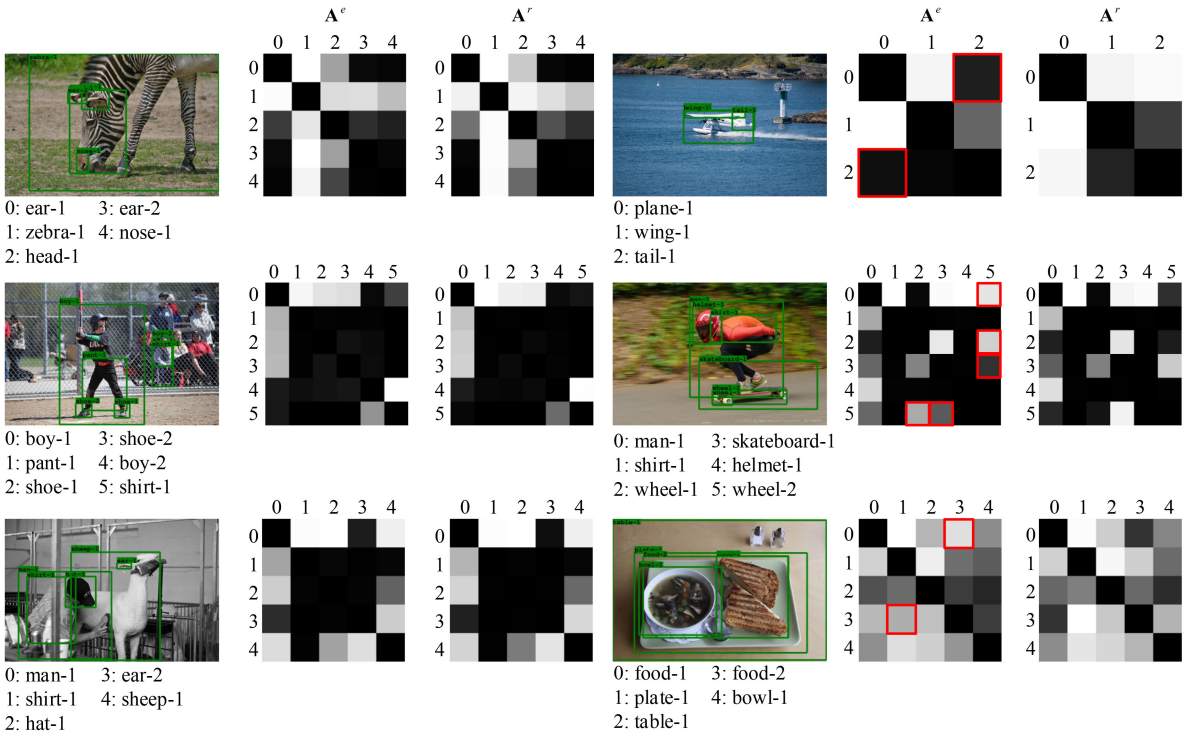



Fig. 5. Some affinity matrix examples produced by R2-Net on *SGCLS*. The small gray square indicates whether there is a relationship between two objects. The lighter the color, the greater the probability. The red bounding boxes indicate visible differences between $\mathbf{A}^e$ and $\mathbf{A}^r$ mentioned in (11) and (13).

TABLE VIII
COMPARISON WITH OTHER METHODS. PERFORMANCE IS COMPUTED WITHOUT GRAPH CONSTRAINTS. THE RESULTS ARE OBTAINED ON THE TEST DATASET. THE SUPERVISED LEARNING STRATEGY IS DENOTED AS SL. THE REINFORCEMENT LEARNING STRATEGY IS DENOTED AS RL

| Learning Strategy | Model | SGDET | | SGCLS | | PREDCLS | |
|---|---|---|---|---|---|---|---|
| | | R@50 | R@100 | R@50 | R@100 | R@50 | R@100 |
| SL | IMP [17] | 22.0 | 27.4 | 43.4 | 47.2 | 75.2 | 83.6 |
| | AE [32] | 9.7 | 11.3 | 26.5 | 30.0 | 68.0 | 75.2 |
| | FREQ+OVERLAP [16] | 28.6 | 34.4 | 39.0 | 43.4 | 75.7 | 82.9 |
| | Motifnet [16] | 30.5 | 35.8 | 44.5 | 47.7 | 81.1 | 88.3 |
| | R2-Net (w/o GCN1) | **32.9** | **38.6** | 45.4 | 48.8 | 81.0 | **88.4** |
| | R2-Net | 30.4 | 36.0 | **46.6** | **49.9** | **81.3** | 88.2 |
| SL+RL | CMAT [47] | 31.6 | 36.8 | 48.6 | 52.0 | 83.2 | 90.1 |

opposite. Also, with the same metric and model, the score of the task *SGCLS* is around 10% higher than the score of the *SGDET* task. These experimental results clearly demonstrate the effectiveness of the accurate bounding boxes and confirm that bounding boxes with noise could lead to a failure of object relation matrix construction, thus further decreasing the role of GCN1. Moreover, for the task of *PREDCLS*, the performance of R2-Net (w/o GCN1) and R2-Net is almost the same. As mentioned in the ablation study (i.e., Section IV-E), the object relation prediction is largely depending on the accuracy of the object labels instead of the object region features. Therefore, with ground-truth object labels, using GCN1 to improve the region features could not considerably improve the performance of the object relation prediction process.

### G. Results and Analysis: Qualitative Results

We show some qualitative examples in Fig. 4 obtained by our R2-Net (w/o GCN1) on the *SGDET*. From the first column, we can see that our model not only predicts relationships in the ground truth but also predicts relationships not in the ground truth, such as (pot-1, has, handle-1), (boat-2, has, dog-1), (tie1, on, shirt-1), and (hair, on, head-1). From the second column, we can see that undetected bounding boxes are a major reason for leading to relation generation failures. For instance, *railing-GT1* and *logo-GT1* in the first row, *tire-GT1* in the second row, *paw-GT1* in the third row and *leg-GT1* in the bottom, are not detected. Therefore, the relations correlated with them are not detected.

We also show some affinity matrices [$\mathbf{A}^e$ and $\mathbf{A}^r$ mentioned in (11) and (13)] in Fig. 5. From examples on the left side, both $\mathbf{A}^e$ and $\mathbf{A}^r$ can properly predict whether there is a relationship between instances. However, in the examples on the right side, the visual link prediction of $\mathbf{A}^r$ is better than $\mathbf{A}^e$. For instance, $\mathbf{A}^r$ can provide correct links of the following object pairs: (plane-1, tail-1), (man-1, wheel-2), (wheel-1, skateboard-1), and (food-2, plate-1). In the right bottom example, *food-2* (i.e., soup and bread) contains *food-1* (i.e., bread), but the relationship category between *food-2* and *food-1* is not contained in the visual genome dataset. Therefore, in the affinity matrix

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



$\mathbf{A}^r$, the value of the object pair (food-1, food-2) is small. The reason why the prediction of $\mathbf{A}^r$ is better than $\mathbf{A}^e$ includes two aspects. First, after the object label refiner, the prediction of the object label is improved. Second, the residual connection allows the object relationship generator to receive more robust features.

## V. Conclusion

In this article, we proposed a novel model for parsing visual scene graphs, namely, R2-Net. It predicts whether there is a relationship between two objects and generates an affinity matrix. GCNs over the affinity matrix aggregate the related object features to the target object features. In this way, the model can integrate the information of related objects to boost the performance of the label prediction. Bi-LSTMs are used to extract the global contexts of objects. By encoding object features with global context and relational information, the relation regularized module can effectively refine the prior labels from faster R-CNN and predict the relationships between objects. Compared with other methods, our model produces more accurate object labels and more robust relationship features, so our model outperforms the state-of-the-art methods on the SGG task. Extensive experiments on the visual genome dataset demonstrate the effectiveness of our method.

## References


[1] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 1–9.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[4] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788.

[5] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NeurIPS*, 2015, pp. 91–99.

[6] J. Johnson *et al.*, "Image retrieval using scene graphs," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 3668–3678.

[7] X. Xu, T. Wang, Y. Yang, L. Zuo, F. Shen, and H. T. Shen, "Cross-modal attention with semantic consistence for image-text matching," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5412–5425, Dec. 2020.

[8] J. Song, H. Zhang, X. Li, L. Gao, M. Wang, and R. Hong, "Self-supervised video hashing with hierarchical binary auto-encoder," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3210–3221, Jul. 2018.

[9] L. Zhu, Z. Huang, Z. Li, L. Xie, and H. T. Shen, "Exploring auxiliary context: Discrete semantic transfer hashing for scalable image retrieval," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5264–5276, Nov. 2018.

[10] C. Ma *et al.*, "Visual question answering with memory-augmented networks," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 6975–6984.

[11] L. Gao, P. Zeng, J. Song, X. Liu, and H. T. Shen, "Examine before you answer: Multi-task learning with adaptive-attentions for multiple-choice VQA," in *Proc. ACM MM*, 2018, pp. 1742–1750.

[12] P. Anderson *et al.*, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 6077–6086.

[13] L. Gao, X. Li, J. Song, and H. T. Shen, "Hierarchical LSTMs with adaptive attention for visual captioning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1112–1131, May 2020.

[14] R. Krishna *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, 2017.

[15] C. Lu, R. Krishna, M. S. Bernstein, and F. Li, "Visual relationship detection with language priors," in *Proc. ECCV*, 2016, pp. 852–869.

[16] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 5831–5840.

[17] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3097–3106.

[18] H. Zhang, Z. Kyaw, S. Chang, and T. Chua, "Visual translation embedding network for visual relation detection," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3107–3115.

[19] Y. Li, W. Ouyang, B. Zhou, J. Shi, C. Zhang, and X. Wang, "Factorizable net: An efficient subgraph-based framework for scene graph generation," in *Proc. ECCV*, 2018, pp. 346–363.

[20] G. Yin *et al.*, "Zoom-Net: Mining deep feature interactions for visual relationship recognition," in *Proc. ECCV*, 2018, pp. 330–347.

[21] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang, "Scene graph generation from objects, phrases and region captions," in *Proc. ICCV*, 2017, pp. 1270–1279.

[22] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph R-CNN for scene graph generation," in *Proc. ECCV*, 2018, pp. 690–706.

[23] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[24] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 580–587.

[25] R. B. Girshick, "Fast R-CNN," in *Proc. ICCV*, 2015, pp. 1440–1448.

[26] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*, 2016, pp. 21–37.

[27] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. ICCV*, 2017, pp. 2980–2988.

[28] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[29] B. Dai, Y. Zhang, and D. Lin, "Detecting visual relationships with deep relational networks," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3298–3308.

[30] Y. Guo, J. Song, L. Gao, and H. T. Shen, "One-shot scene graph generation," in *Proc. ACM MM*, 2020, pp. 3090–3098.

[31] A. Bordes, N. Usunier, A. García-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. NeurIPS*, 2013, pp. 2787–2795.

[32] A. Newell and J. Deng, "Pixels to graphs by associative embedding," in *Proc. NeurIPS*, 2017, pp. 2168–2177.

[33] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016. [Online]. Available: arxiv.abs/1609.02907.

[34] M. S. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proc. ESWC*, 2018, pp. 593–607.

[35] D. Marcheggiani and I. Titov, "Encoding sentences with graph convolutional networks for semantic role labeling," in *Proc. EMNLP*, 2017, pp. 1506–1515.

[36] S. Qi, W. Wang, B. Jia, J. Shen, and S. Zhu, "Learning human-object interactions by graph parsing neural networks," in *Proc. ECCV*, 2018, pp. 407–423.

[37] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[38] L. He, K. Lee, M. Lewis, and L. Zettlemoyer, "Deep semantic role labeling: What works and what's next," in *Proc. ACL*, 2017, pp. 473–483.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] Y. Zhang, G. Chen, D. Yu, K. Yao, S. Khudanpur, and J. R. Glass, "Highway long short-term memory RNNS for distant speech recognition," in *Proc. ICASSP*, 2016, pp. 5755–5759.

[41] B. Yang, W. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," in *Proc. ICLR*, 2015.

[42] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. ACL*, 2014, pp. 1532–1543.

[43] R. Herzig, M. Raboh, G. Chechik, J. Berant, and A. Globerson, "Mapping images to scene graphs with permutation-invariant structured prediction," in *Proc. NeurIPS*, 2018, pp. 7211–7221.


Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



[44] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, p. 533, 1986.
[45] S. J. Hwang *et al.*, "Tensorize, factorize and regularize: Robust visual relationship learning," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1014–1023.
[46] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 6619–6628.
[47] L. Chen, H. Zhang, J. Xiao, X. He, S. Pu, and S. Chang, "Counterfactual critic multi-agent training for scene graph generation," in *Proc. ICCV*, 2019, pp. 4612–4622.

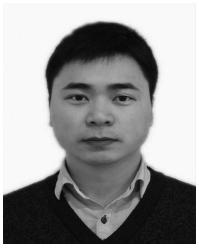
**Yuyu Guo** is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China.

He is currently working on image understanding, image/video captioning, and scene graph generation.

**Lianli Gao** (Member, IEEE) received the Ph.D. degree in information technology from the University of Queensland, Brisbane, QLD, Australia, in 2015.

She is currently a Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. She is focusing on integrating natural language for visual content understanding.

Prof. Gao was the winner of the IEEE TRANSACTIONS ON MULTIMEDIA 2020 Prize Paper Award, the Best Student Paper Award in the Australian Database Conference, Australia, in 2017, the IEEE TCMC Rising Star Award in 2020, and the ALIBABA Academic Young Fellow.



**Jingkuan Song** (Senior Member, IEEE) received the Ph.D. degree in information technology from the University of Queensland, Brisbane, QLD, Australia, in 2014. He is currently a Professor with the University of Electronic Science and Technology of China, Chengdu, China. His research interests include large-scale multimedia retrieval, image/video segmentation and image/video understanding using hashing, graph learning, and deep learning techniques.

Prof. Song was the winner of the Best Paper Award in the International Conference on Pattern Recognition, Mexico, in 2016; the Best Student Paper Award in Australian Database Conference, Australia, in 2017; and the Best Paper Honorable Mention Award, Japan, in 2017. He has been an AC/SPC/PC Member of the IEEE Conference on Computer Vision and Pattern Recognition for the term 2018 to 2021, and so on.

**Peng Wang** received the Ph.D. degree from the University of Queensland, Brisbane, QLD, Australia, in 2017.

He was a Research Fellow with the Australia Institute of Machine Learning, University of Adelaide, Adelaide, SA, Australia. He is currently a Lecturer with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia. His research works have been published in major computer vision journals and conferences, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *International Journal of Computer Vision*, IEEE TRANSACTIONS ON IMAGE PROCESSING, CVPR, and AAAI. His research interests include computer vision and deep learning, with a focus on low-shot classification, long-tail classification, and visual reasoning.

**Nicu Sebe** (Senior Member, IEEE) received the Ph.D. degree from Leiden University, The Netherlands, in 2001. He is a Professor with the University of Trento, Trento, Italy, leading the research in the areas of multimedia information retrieval and human behavior understanding.

Prof. Sebe was the General Co-Chair of the IEEE FG Conference 2008 and ACM Multimedia 2013, and the Program Chair of the International Conference on Image and Video Retrieval in 2007 and 2010, ACM Multimedia in 2007 and 2011, and ICCV 2017 and ECCV 2016. He was the General Chair of ACM ICMR 2017. He is a Fellow of IAPR.

**Heng Tao Shen** (Senior Member, IEEE) received the B.Sc. (First Class Hons.) and Ph.D. degrees in computer science from the Department of Computer Science, National University of Singapore, Singapore, in 2000 and 2004, respectively.

He is currently a Professor, the Dean of School of Computer Science and Engineering, the Executive Dean of the Artificial Intelligence Research Institute, and the Director of Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China. He has published over 250 peer-reviewed papers. His current research interests include multimedia search, computer vision, artificial intelligence, and big data management.

Prof. Shen received seven best paper awards from international conferences, including the Best Paper Award from ACM Multimedia 2017 and Best Paper Award-Honourable Mention from ACM SIGIR 2017. He has served as the General Co-Chair for ACM Multimedia 2021 and the Program Committee Co-Chair for ACM Multimedia 2015. He is an Associate Editor of *ACM Transactions of Data Science*, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.

**Xuelong Li** (Fellow, IEEE) is a full professor with School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on September 04,2021 at 09:13:05 UTC from IEEE Xplore. Restrictions apply.