mathematics

MDPI

*Article*

# Region Collaborative Network for Detection-Based Vision-Language Understanding

**Linyan Li [1,†], Kaile Du [2,†], Minming Gu [2], Fuyuan Hu [2,*] and Fan Lyu [3]**

[1] Suzhou Institute of Trade & Commerce, Suzhou 215009, China
[2] Electronic & Information Engineering, Suzhou University of Science and Technology, Suzhou 215009, China
[3] College of Intelligence and Computing, Tianjin University, Tianjin 300000, China
* Correspondence: fuyuanhu@mail.usts.edu.cn
† These authors contributed equally to this work.

**Abstract:** Given a query language, a Detection-based Vision-Language Understanding (DVLU) system needs to respond based on the detected regions (i.e.,bounding boxes). With the significant advancement in object detection, DVLU has witnessed great improvements in recent years, such as Visual Question Answering (VQA) and Visual Grounding (VG). However, existing DVLU methods always process each detected image region separately but ignore that they were an integral whole. Without the full consideration of each region's context, the image's understanding may contain more bias. In this paper, to solve the problem, a simple yet effective Region Collaborative Network (RCN) block is proposed to bridge the gap between independent regions and the integrative DVLU task. Specifically, the Intra-Region Relations (IntraRR) inside each detected region are computed by a position-wise and channel-wise joint non-local model. Then, the Inter-Region Relations (InterRR) across all the detected regions are computed by pooling and sharing parameters with IntraRR. The proposed RCN can enhance the features of each region by using information from all other regions and guarantees the dimension consistency between input and output. The RCN is evaluated on VQA and VG, and the experimental results show that our method can significantly improve the performance of existing DVLU models.

**Keywords:** detection-based vision-language understanding; region collaborative network; non-local network

**MSC:** 68T07



**Citation:** Li, L.; Du, K.; Gu, M.; Hu, F.; Lyu, F. Region Collaborative Network for Detection-Based Vision-Language Understanding. *Mathematics* **2022**, *10*, 3110. https://doi.org/10.3390/math10173110

Academic Editors: Xiangtao Zheng, Jinchang Ren and Ling Wang

Received: 1 August 2022
Accepted: 24 August 2022
Published: 30 August 2022



## 1. Introduction

Visual object detection has been studied for years and has been applied to many applications such as smart city, robot vision, consumer electronics, security, autonomous driving, human–computer interaction, content-based image retrieval, intelligent video surveillance, and augmented reality [1–3]. Recently, many researchers found that vision-language understanding tasks based on detected regions or called Detection-based Vision-Language Understanding (DVLU) can achieve better performance than many traditional vision-language understanding methods that directly use the pixel-based image information [4–7]. For instance, the key to the success of the winning entries to the Visual Question Answering (VQA) challenge [8–12] is the use of detected regions. Moreover, the success of visual object detection leads to many new DVLU tasks, such as Visual Grounding [13–17], Visual Relationship Detection [18,19] and Scene Graph Generation [20–24]. With significant advancements made in the area of object detection, DVLU and its related tasks have attracted much attention in recent years. As shown in Figure 1, in a smart city, a monitor may shoot a picture from a crowded street, and a DVLU system can answer the human question and give the following response. For example, when given a query question "How many

people are in the picture?", a DVLU system can provide the correct answer via the detected regions.

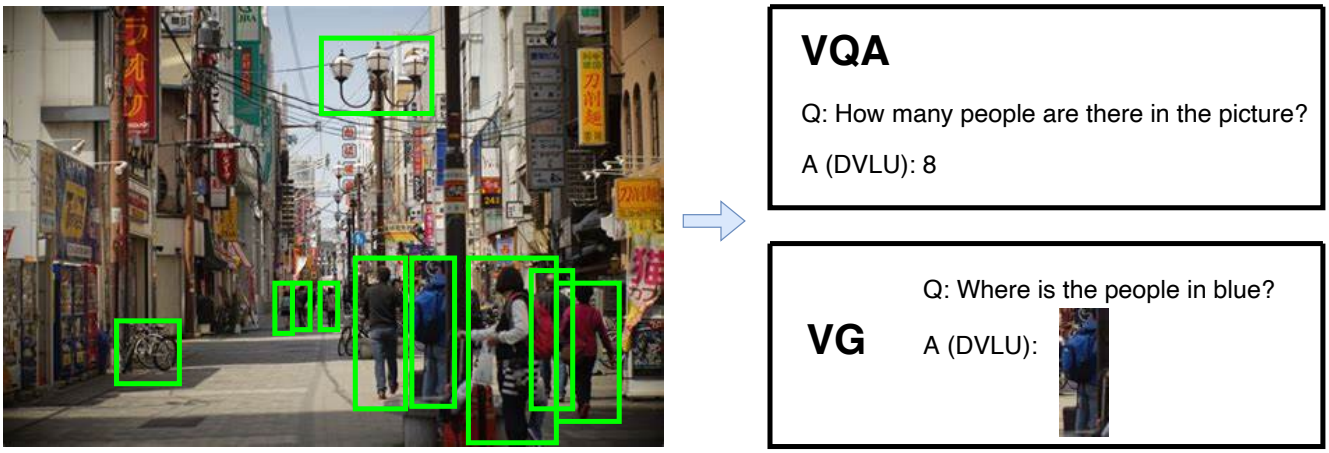


**Figure 1.** A DVLU system in a smart city. Specifically, VQA needs to answer with natural language while VG needs to answer with a grounded region.

A DVLU task is defined as that given several detected regions (hereinafter, all regions refer to detected regions or bounding boxes for simplicity) from a nature image, a model automatically recognizes the semantic patterns or concepts inside the image. Specifically, a DVLU model first accepts some region inputs from an off-the-shelf object detection model (detector). By applying any well-designed model (i.e., a deep neural network or DNN), these detected regions can be treated as finer features or bottom-up attentions [6] of the original image. Then, each region is fed to the DVLU model as a kind of meaningful information to help understand images at a high level. Unlike usual image understanding, DVLU is based on multiple detected or labeled regions, which makes DVLU receive more fine-grained information than non-region image understanding.

Nevertheless, when facing numerous independent detected regions in an image, existing DVLU methods always ignore their distributions and relationships, which is inadequate for different regions consisting of a real-world scene. Furthermore, it is insufficient to separately reprocess each region for features after detecting them while ignoring that they were an integral whole and discarding their original cooperation. This motivates our new design of collaboratively processing all the regions for DVLU, which will help a DVLU task make full use of regions to collaborate and enhance each other.

In this paper, we propose a simple yet effective add-in block of Region Collaborative Network (RCN) that makes every region collaborate with others in DVLU. With RCN, the feature of a region is enhanced by all other regions in the image. As shown in Figure 2, in a DVLU pipeline, the RCN block can be easily inserted by keeping the dimension consistent. First, the RCN block receives the feature of regions as input. For each region feature, we compute Intra-Region Relation (IntraRR) across positions and channels to achieve self-enhancement. Then, the Inter-Region Relation (InterRR) is applied to all regions across positions and channels. To reduce redundant connections, we apply a pooling and sharing strategy relative to IntraRR. Finally, a series of the refined feature of regions are obtained by linear embedding. In general, we can easily add an RCN block to any DVLU model to process all the regions collaboratively. The proposed RCN is put into two DVLU tasks in the experiment: Visual Question Answering and Visual Grounding. The experimental results show that the proposed RCN block can significantly improve the performance against the baseline models.

Our contributions are three-fold:

1. In RCN, we propose to build Intra-Region Relation (IntraRR) across positions and channels to achieve self-enhancement.

2. We propose to build Inter-Region Relation (InterRR) to all regions across positions and channels for relationships among regions.
3. We evaluate the effectiveness of RCN on VQA and VG, and the results show the proposed method can improve the performance of existing DVLU methods.

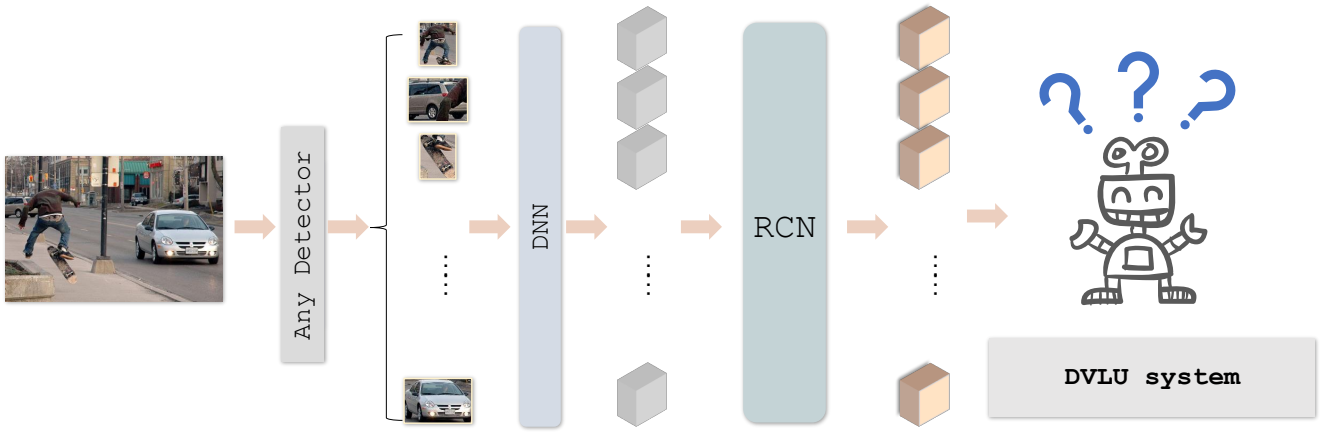


**Figure 2.** Region Collaborative Network (RCN) can be inserted into a Detection-based Vision-Language Understanding (DVLU) pipeline. Any object detector can be used to generate regions.

## 2. Related Work

### *2.1. Detection-Based Vision-Language Understanding*

Detection-based Vision-Language Understanding (DVLU) attracted much attention in recent years due to the significant technological advancement of visual object detection [1,25–28]. With semantically labeled regions, DVLU methods can mine deeper-level information more effectively than those based only on pixel-based images. DVLU methods have been applied to many high-level vision-language tasks, such as VQA [6,7,13,29,30] and VG [13,31,32]. In VQA, Anderson et al. [6] build an attention model on detected bounding boxes for image captioning and visual question answering. Hinami et al. [33] incorporate the semantic specification of objects and intuitive specification of spatial relationships for detection-based image retrieval. The up–down [8] model is based on the principle of a joint embedding of the input question and image, followed by a multi-label classifier over a set of candidate answers. In pythia [9], they demonstrate that by making subtle but important changes to the model architecture and the learning rate schedule, fine-tuning image features, and adding data augmentation, pythia can significantly improve the performance of the up–down model. In VG, Deng et al. [13] propose to build the relations among regions, language, and pixels using an accumulated attention mechanism. Lyu et al. [32] propose constructing the graph relationships among all regions and using a graph neural network to enhance the region feature. In [34], the model exploits the reciprocal relation between the referent and context, either of them influences the estimation of the posterior distribution of the other, and thereby the search space of context can be significantly reduced. However, most of these methods do not fully use the relationship among regions, and some only consider the relative spatial relation of regions. In general, these methods ignore the cross-region problem. This may isolate each region from the whole scene and negatively affect performance.

### *2.2. Region Relation Network*

Relationships among objects or regions have been studied for years. Some work is based on hyperspectral imaging [35–38] and some on deep learning [39–43]. One way is to study the explicit relationship between objects. It usually starts with detecting every object pair with the relationship in a given image [18,19]. After that, by combining detected relationship triplets (⟨subject, relationship, object⟩), an image can be transformed into a scene graph, where each node represents the region and each directed edge denotes

the relationship between two regions [4,20–22]. Unfortunately, it is still unclear how to effectively use this structural topology for many end-user tasks, although it is based upon a large number of labeled relationships. Another way is to focus on building a latent relationship between objects. Zagorukyo et al. [44] explore and study several neural network architectures that compare image patches. Tseng et al. [45] was inspired by the non-local network [46] and proposed to build non-local ROI for cross-object perception. This paper is different from [45] in that the proposed RCN: (1) can be inserted into any DVLU model while keeping dimension consistency without modifying the original model; (2) computes intra-region and Inter-Region Relations in terms of both spatial positions and channels; and (3) applies to fully connected region features.

## 3. Method

### *3.1. Review of Non-Local Network*

Recently, the non-local network has been verified as effective to improve recognition by modeling the long-range region relation in a network [45–48] in which a self-attention mechanism is used to form an attention map. For each position in the feature (say, query point), the non-local network first computes its pairwise relations to all other positions and then aggregates the features of all positions by weighted sum. The non-local network inspired us to construct the region relation in our RCN. The RCN builds not only Intra-Region Relation, like [45–48] but also Inter-Region Relation. Following [46,47], a generic non-local operation in DNNs can be defined as

$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{j=1}^{D} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j), \tag{1}$$

where the pairwise function $f(\cdot)$ computes a scalar that reflects the correlation between the target position and every other position in the feature map $\mathbf{x} \in \mathbb{R}^{H \times W \times d}$, and $g(\cdot)$ is a mapping function. $D = HW$ denotes all positions in a feature. This representation is normalized by a factor $\mathcal{C}(\mathbf{x}) = \sum_{j=1}^{D} f(\mathbf{x}_i, \mathbf{x}_j)$. Here, we name $\mathbf{x}_i$ a query point, while calling each $\mathbf{x}_j$ a connection point. The non-local network is a simple but effective solution to extend the local processing of DNN's convolutional layer for any position to a non-local pattern. It is even capable of capturing long-range dependencies between positions.

### *3.2. Region Collaborative Network*

Given an image $I$, a set of detected regions $\mathcal{O} = \{o^1, \ldots, o^N\}$ is obtained by leveraging an object detector such as Faster R-CNN [25] and YOLO [26], or even from manual labeling. By using a feature extractor (for example RoI-pooling/aligning or linear embedding), regions are represented by fixed-size features $\mathcal{X} = \{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$. Here, the Region Collaborative Network block (RCN block) is added in any DVLU task denoted as

$$\hat{\mathbf{y}}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{n=1}^{N} \sum_{j=1}^{D} f(\mathbf{x}_i, \mathbf{x}_j^n) g(\mathbf{x}_j^n). \tag{2}$$

The differences between RCN and the original non-local network are: (1) The input to our model is the high-level feature of all detected regions of an image; (2) The pairwise function $f(\cdot)$ computes correlations not only inside one region but across multiple regions; (3) The normalization factor is distributed through all positions of all regions $\mathcal{C}(\mathbf{x}) = \sum_{n=1}^{N} \sum_{j=1}^{D} f(\mathbf{x}_i, \mathbf{x}_j^n)$. Specifically, as shown in Figure 3, RCN holds two main operations, i.e.,Intra-Region Relation (IntraRR) and Inter-Region Relation (InterRR). The IntraRR is particularly designed for one region's feature, which enhances each position and channel by other positions and channels. The InterRR is adapted to the features from multiple regions, in which other regions enhance each position and channel in one region.

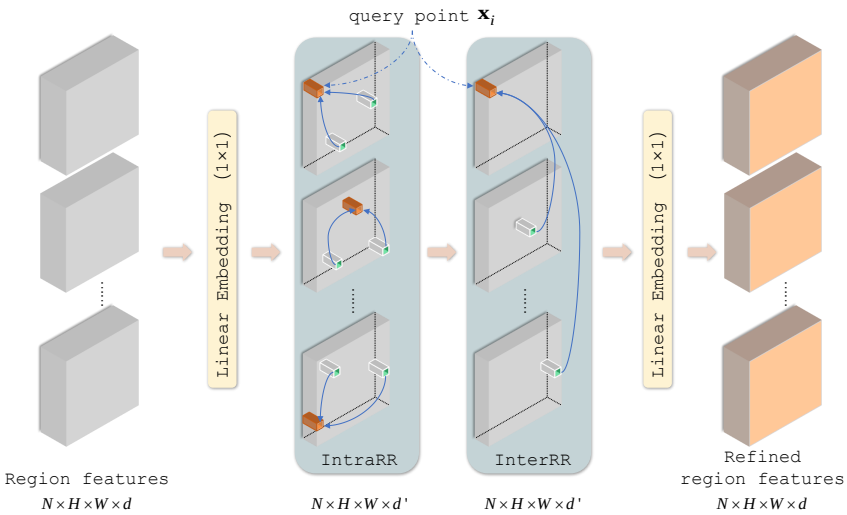


**Figure 3.** The proposed RCN receives the input from the features of multiple detected regions while enforcing the same dimension for the input and output. RCN consists of two main parts, Intra-Region Relation (IntraRR) and Inter-Region Relation (InterRR).

#### 3.2.1. Intra-Region Relation

First, we argue that each region that is framed represents a conception or a goal. Thus, the feature directly extracted from a region by a DNN can be seen as a reasonable representation. To further capture the long dependencies inside the region, we construct the Intra-Region Relation (IntraRR). A simple and straightforward way to grab IntraRR is by treating each region as an independent patch and applying non-local operations directly. However, the operation computes relationship only over spatial distribution (or position-wise), while CNN features are naturally spatial and channel-wise, which may result in losing channel-wise dependencies. Hence, as shown in Figure 4, a position-wise and a channel-wise non-local operation are built, respectively, on the high-level feature tensor $\mathbf{x} \in \mathbb{R}^{H \times W \times d}$ of every region

$$\mathbf{y} = \mathbf{y}^{\text{position}} + \mathbf{y}^{\text{channel}} + \mathbf{x}, \tag{3}$$

where for $\mathbf{y}^{\text{position}}$ and $\mathbf{y}^{\text{channel}}$, D is equal to $HW$ and $d$, respectively, in Equation (1), which is the same with the normalization. This operation can be regarded as a feature refinement. In other words, by considering dependencies across positions and channels, a region feature guarantees a dense self-correlation. Note, if the region feature is from a fully connected layer, we do not compute ItraRR, since the fully connected layer is already a suitable embedding method that adds the linear combination for each value.

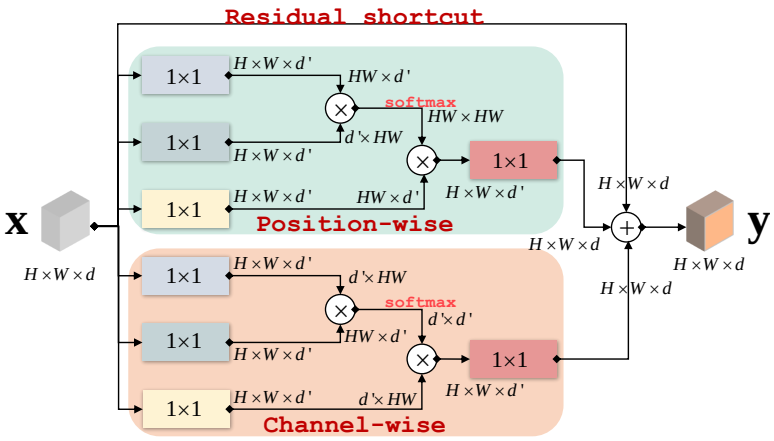


**Figure 4.** Intra-Region Relation (IntraRR) for convolutional region features: given the feature map of a region $\mathbf{x} \in \mathbb{R}^{H \times W \times d}$, IntraRR computes the relations in position-wise and channel-wise.

#### 3.2.2. Inter-Region Relation

In the IntraRR, each region is self-correlated. As mentioned, it is also worth constructing the relation across regions because their full cooperation creates a meaningful scene. Similarly, for the feature set $\mathcal{X} = \{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ of all detected regions $\mathcal{O} = \{o_1, \ldots, o_N\}$, we are looking forward to construct a mapping $\mathbf{y}^n = h(\mathbf{x}^n, \mathcal{X})$, where $N$ is the number of regions. Specifically, like intra-proposal relation, every location or channel is supposed to correspond to other regions. Thus, as shown in Figure 5, the feature of the $n$-th region can be obtained by

$$\hat{\mathbf{y}}^n = \hat{\mathbf{y}}^{n(\text{position})} + \hat{\mathbf{y}}^{n(\text{channel})} + \mathbf{x}^n. \tag{4}$$

To compute $\hat{\mathbf{y}}^{n(\text{position})}$ and $\hat{\mathbf{y}}^{n(\text{channel})}$, Equation (2) is applied, where their $D$ is equal to $HW$ and $d$, respectively. However, the dense connections ($(ND)^2$) for position and channel are much redundant and result in learning obstacles and even failure training. A simple strategy is used to reduce redundant connection by *pooling and sharing* (PAS) with IntraRR. We suppose using only one query point to represent a detected region is possible. For convenience, the average pooling in the feature of a region is used to obtain the query point, and the $n$-th region can be represented by

$$\mathbf{z}^n = \frac{1}{\mathcal{C}(\mathbf{x}^n)} \sum_{j=1}^{D} f(\text{pool}(\mathbf{x}^n), \mathbf{x}_j^n) g(\mathbf{x}_j^n), \tag{5}$$

where the correlation pairwise function $f(\cdot)$ and the mapping function $g(\cdot)$ share parameters with IntraRR. Thus, Equation (2) can be rewritten as

$$\hat{\mathbf{y}}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{n=1}^{N} f(\mathbf{x}_i, \mathbf{z}^n) g(\mathbf{z}^n), \tag{6}$$

where $\mathcal{C}(\mathbf{x}) = \sum_{n=1}^{N} f(\mathbf{x}_i, \mathbf{z}^n)$. Thus, the connections are reduced to $N^2 D$.

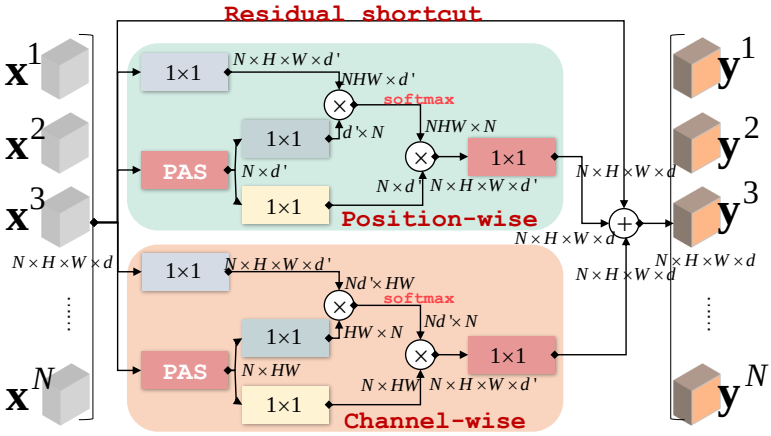


**Figure 5.** Inter-Region Relation (InterRR) for convolutional region features: given the features of a number of detected regions $\mathcal{X} = \{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$, InterRR also computes the relations in position-wise and channel-wise but across regions.

#### 3.2.3. RCN for Fully Connected Layer

In some models, detected regions are represented by the vector from a fully connected layer in DNN. In this case, the IntraRR is not computed because the fully connected layer is already an excellent dense embedding method that adds the linear combination for a region. However, the InterRR information is still computed for each region in this situation.

In this case, the problem is simplified as $D = 1$ and without position and channel element. Therefore, for a query point vector $\mathbf{x}^n \in \mathbb{R}^{1\times d}$, the InterRR is computed as

$$\hat{\mathbf{y}}^n = \frac{1}{\mathcal{C}(\mathbf{x}^n)} \sum_{m=1}^{N} f(\mathbf{x}^n, \mathbf{x}^m) g(\mathbf{x}^m), \tag{7}$$

where $\mathcal{C}(\mathbf{x}^n) = \sum_{m=1}^{N} f(\mathbf{x}^n, \mathbf{x}^m)$. Inter-Region Relation (InterRR) for fully connected region features is shown in Figure 6.

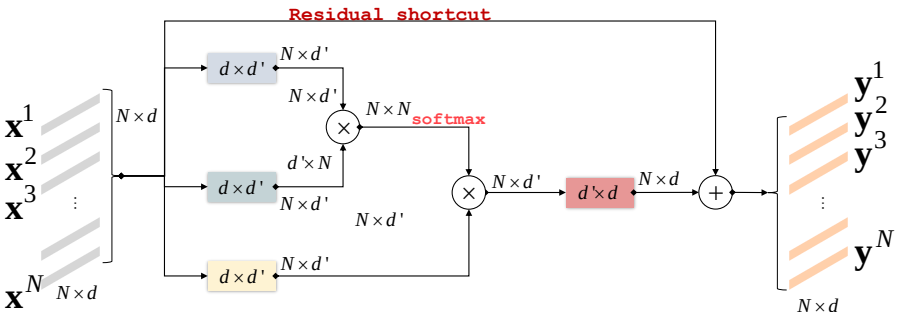


**Figure 6.** Inter-Region Relation (InterRR) for fully connected region features: when the feature of regions are from a fully connected layer, they will be formed as several vectors; the InterRR can be computed as a simplified version of InterRR of convolutional features.

## 4. Experiments

The proposed RCN is implemented in two representative tasks corresponding to three main different levels of DVLU, i.e.,Visual Question Answering (VQA) and Visual Grounding (VG).

### *4.1. Implementation Detail*

Our tracker is implemented in Python, using PyTorch. It runs on a GeForce GTX TITAN X GPU. Following [8] and [9], $N$ regions are represented by $2048D$ features from the fc6 layer of ResNeXt [49], where $K <= 100$. Our model is built based on pythia [9] and has the same setting as the best model (without ensemble) presented by their project (https://github.com/facebookresearch/pythia, accessed on 31 July 2022). As an add-in block, our implementation is based on VC (https://github.com/yuleiniu/vc/, accessed on 31 July 2022), and an RCN block is inserted into the neural network. All other hyper-parameters are set to the same as VC. The evaluation metric is accuracy, which means the rate selects the correct regions with the guides from query sentences. An RCN block is inserted before the modal combining operation, combining the image and question features. In other words, it is expected that every combination should consider the picked region and the relationships between this region and others. We use accuracy as our evaluation metric for both VQA and VG, like [11,12,16,17]. The VQA score is calculated by comparing the inferred answer to 10 ground-truth answers. In our method, the running time of the VQA task is about 260 ms, and the running time of the VG task is about 320 ms.

### *4.2. Bottom-Up Visual Question Answering*

The recipe of the winning entries to the VQA challenge [8,9] is the use of the object detector (Faster R-CNN [25] and FPN [28]) to extract detected regions as bottom-up attention. By applying top-down attention to each region in the image and fusing every feature through the Hadamard product, it is easy to predict the answer score for each question. Their glorious prizes show that detected regions of images can significantly improve the performance of traditional methods that only use an image. However, it is a pity that their models never consider the cross-region relation but directly use regions' linear combination to represent an image. Although it is helpful to assign a factor to every region feature by linearly embedding the concatenation of the question feature and region feature, we think

it should consider other regions when computing the relationship between the question and one region. Thus, the proposed RCN is first evaluated on VQA.

#### 4.2.1. Dataset

We evaluate our method on the popular VQA 2.0 dataset. VQA 2.0 is a dataset containing open-ended questions about images. These questions require an understanding of vision, language, and commonsense knowledge to answer. The dataset contains 265,016 images and at least 3 questions (5.4 questions on average) per image. Each question has 10 ground-truth answers. We use split train and validation data as the training set and test-dev as the test set.

#### 4.2.2. Results

Our method is compared with several famous detection-based VQAs, including HieCoAtt-ResNet [50], VQA-Machine-ResNet [51], Ma et al. -ResNet [8], MCB-ResNet [52], up–down [8], and pythia [9]. The experimental results are shown in Table 1. Based on the champion of the VQA challenge 2018, pythia, we insert in an RCN block and train the model from scratch. Compared with a vanilla pythia, an improvement in answer accuracy can be observed (from 68.49% to 68.56%). The results show the proposed RCN can improve further in contrast to the baseline pythia. This shows the proposed RCN does help the VQA task. We believe that before combining image and question features, a question should not only consider but inspect each region independently. A question may come to the relationships among regions, such as "What does the man hold?"—"frisbee", where the regions of man and frisbee and their relationship should be considered.

**Table 1.** Single-model VQA 2.0 performance in %. Based on the champion of the VQA challenge 2018, pythia, we insert an RCN block and train the model from scratch.

| Model | Test-Dev |
|---|---|
| HieCoAtt-ResNet [50] | 61.80 |
| VQA-Machine-ResNet [51] | 63.10 |
| Ma et al. -ResNet [8] | 63.80 |
| MCB-ResNet [52] | 64.70 |
| up-down [8] (Champion of VQA Challenge 2017) | 65.32 |
| pythia [9] (Champion of VQA Challenge 2018) | 68.49 |
| pythia + RCN | **68.56** |

### *4.3. Visual Grounding*

Visual Grounding (VG) is a challenging cross-modal task that automatically uses text description to localize an object of interest from an image. The common design of the present VG is formulating the task of selecting the best regions from a set of given proposals [13,53–57] by fusing features from multiple modalities. VC [34] is one of the state-of-the-art methods of VG. It introduces the variational context model where the reciprocity can interpret the variational lower bound between the referent and context.

#### 4.3.1. Dataset

Three popular VG dataset are used, i.e., ReferCOCO, ReferCOCO+, and ReferCOCOg [58]. The three datasets are based on MS-COCO [59]. In ReferCOCO, the queries are short phrases, while in ReferCOCO+ and ReferCOCOg, their queries are normally declarative sentences, short and long. We have the same data splits as [14,56,60], and the detected bounding boxes are from the ground truth. ReferCOCO and ReferCOCO+ are split into train, val, testA, and testB, where split testA contains only people, while other objects are in split testB. In contrast, ReferCOCOg has no test split.

#### 4.3.2. Results

As shown in Table 2, we compare several state-of-the-art VG methods. VG comparisons on ReferCOCO, ReferCOCO+, and ReferCOCOg. VC (ours) means the results of the author's code in our environment are retrained. By inserting the RCN block, a VC model can achieve better performances than the baseline, and even surpass the state-of-the-art methods in some splits. VC achieves the best on the split testB of ReferCOCO and ReferCOCO+ and the split val on ReferCOCOg. By adding the RCN block, VC+RCN obtains better results, where 2.39 and 0.63 improvements compared with the claimed results of VC can be observed on the split testA of ReferCOCO and ReferCOCO+ and 0.4 and 0.45 on testB. We find a decline in our implemented VC on three datasets compared with the claimed VC. Compared with the VC that retrained the author's code in our environment (70.21%), the result of VC+RCN achieves a clear performance improvement. This demonstrates the effectiveness of the proposed RCN; when combined with a VG model such as VC, each region can be enhanced by other regions and further benefit the VG model. This will be more notable when coming up with a complex sentence containing many references. For instance, for "the man holding a frisbee behind a tree", the regions of frisbee and tree will provide more critical information about the feature of the region of the man.

**Table 2.** VG Comparisons of ReferCOCO, ReferCOCO+, and ReferCOCOg (accuracy in %). VC (ours) means the results of the author's codes in our environment are retrained. The bold type indicates the state-ot-the-art performance.

| **Methods** | **Regions** | **ReferCOCO** | | | **ReferCOCO+** | | | **ReferCOCOg** |
|---|---|---|---|---|---|---|---|---|
| | | **Val** | **TestA** | **TestB** | **Val** | **TestA** | **TestB** | **Val** |
| Luo et al. [53] | gt | - | 74.14 | 71.46 | - | 59.87 | 54.35 | 63.39 |
| Luo et al. (w2v) [53] | gt | - | 74.04 | 73.43 | - | 60.26 | 55.03 | 65.36 |
| speaker+listener+reinforcer+MMI [61] | gt | 79.56 | 78.95 | 80.22 | 62.26 | 64.60 | 59.62 | 72.63 |
| A-ATT-r4 [13] | gt | 81.27 | 81.17 | 80.01 | 65.56 | **68.76** | 60.63 | 73.18 |
| MAttN [14] | gt | 80.94 | 79.99 | 82.30 | 63.07 | 65.04 | 61.77 | 73.08 |
| VC [34] | gt | - | 78.98 | 82.39 | - | 62.56 | 62.90 | 73.98 |
| VC (Ours) | gt | - | 78.08 | 82.49 | - | 60.08 | 61.78 | 70.21 |
| VC+RCN | gt | **82.03** | **81.37** | **82.79** | **66.71** | 63.19 | **63.35** | **74.10** |

To reduce the computation, the input feature ($d$) is embedded into a small-sized one ($d'$). We also evaluate how much the embedding size of the embedding feature is more suitable for RCN in VG. As shown in Table 3, the performances on ReferCOCO with different embedding sizes (from 256 to 2048) of RCN can be seen. It is easy to see that too large or too small an embedding size is not good for the performance, and a proper size (1024) can make a VC model better. A large embedding size means more information but more parameters, and less size is the contrary. We find a small embedding size indeed leads to less computation due to the fewer parameters but results in a slight decline in performance. Moreover, a larger embedding size does not bring about more robust performance but is even worse. A proper embedding size such as 1024 can make the model gain either less computation or high performance (81.37% and 82.79%).

**Table 3.** Performances on ReferCOCO with different embedding sizes (from 256 to 2048) of RCN. The bold type indicates the state-ot-the-art performance.

| **Methods** | **Embedding Size $d'$** | **ReferCOCO** | |
|---|---|---|---|
| | | **TestA** | **TestB** |
| VC [34] | - | 78.98 | 82.39 |
| VC (Ours) | - | 78.08 | 82.49 |
| VC+RCN | 256 | 79.12 | 82.40 |
| VC+RCN | 512 | 80.07 | 82.14 |
| VC+RCN | 1024 | **81.37** | **82.79** |
| VC+RCN | 2048 | 80.95 | 82.12 |

## 5. Conclusions

Existing Detection-based Vision-Language Understanding (DVLU) methods always process each input image's detected region separately but ignore that they were an integral whole. This paper proposes a Region Collaborative Network (RCN) to address this problem and an easy-to-add-in RCN block that can be inserted into any DVLU method. Specifically, Intra-Region Relations (IntraRR) are computed inside any detection region by a position-wise and channel-wise joint non-local model. Then, the Inter-Region Relations (InterRR) across all detected regions are computed by pooling and sharing parameters with IntraRR. The proposed RCN makes each region enhanced by all other regions without dimension changes. The proposed RCN can significantly fill the gap between independent regions and DVLU tasks. The RCN is evaluated in two classic DVLU tasks, i.e.,visual question answering and visual grounding. Experimental results show that our method can improve the performance of an existing region-based model. The limitation of our paper is that currently, in our RCN, the regions need to be annotated by the feed, which brings additional human annotation costs. We will combine the one-stage detection method in future work to realize end-to-end model training.

**Author Contributions:** L.L.: conceptualization, methodology, and investigation; K.D.: software, writing—original draft, and visualization; M.G.: supervision, writing, review, and editing; F.H.: supervision, writing, review, and editing. F.L.: review, and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Natural Science Foundation of China (No. 61876121), the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant (Nos. 19KJB520054, 20KJB510026), the Research Fund of Suzhou Institute of Trade and Commerce (No. KY-ZRA1805), and Science and Technology Project of Suzhou Water Conservancy (No. 2020007).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** We would like to thank constructive and valuable suggestions for this paper from the experienced reviewers.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: A Survey. *arXiv* **2018**, arXiv:1809.02165.
2. Peng, C.; Weiwei, Z.; Ziyao, X.; Yongxiang, T. Traffic Accident Detection Based on Deformable Frustum Proposal and Adaptive Space Segmentation. *Comput. Model. Eng. Sci.* **2022**, *130*, 97–109.
3. Yunbo, R.; Hongyu, M.; Zeyu, Y.; Weibin, Z.; Faxin, W.; Jiansu, P.; Shaoning, Z. B-PesNet: Smoothly Propagating Semantics for Robust and Reliable Multi-Scale Object Detection for Secure Systems. *Comput. Model. Eng. Sci.* **2022**, *132*, 1039–1054.
4. Johnson, J.; Krishna, R.; Stark, M.; Li, L.J.; Shamma, D.; Bernstein, M.; Fei-Fei, L. Image Retrieval using Scene Graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3668–3678.
5. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
6. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6077–6086.
7. Fu, K.; Jin, J.; Cui, R.; Sha, F.; Zhang, C. Aligning where to see and what to tell: Image captioning with region-based attention and scene-specific contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2321–2334. [CrossRef] [PubMed]
8. Teney, D.; Anderson, P.; He, X.; van den Hengel, A. Tips and Tricks for Visual Question Answering: Learnings from the 2017 Challenge. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4223–4232.
9. Jiang, Y.; Natarajan, V.; Chen, X.; Rohrbach, M.; Batra, D.; Parikh, D. Pythia v0. 1: The Winning Entry to the VQA Challenge 2018. *arXiv* **2018**, arXiv:1807.09956.

10. Biten, A.F.; Litman, R.; Xie, Y.; Appalaraju, S.; Manmatha, R. Latr: Layout-aware transformer for scene-text vqa. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 16548–16558.
11. Cascante-Bonilla, P.; Wu, H.; Wang, L.; Feris, R.S.; Ordonez, V. Simvqa: Exploring simulated environments for visual question answering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 5056–5066.
12. Gupta, V.; Li, Z.; Kortylewski, A.; Zhang, C.; Li, Y.; Yuille, A. Swapmix: Diagnosing and regularizing the over-reliance on visual context in visual question answering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 5078–5088.
13. Deng, C.; Wu, Q.; Wu, Q.; Hu, F.; Lyu, F.; Tan, M. Visual Grounding via Accumulated Attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7746–7755.
14. Yu, L.; Lin, Z.; Shen, X.; Yang, J.; Lu, X.; Bansal, M.; Berg, T.L. Mattnet: Modular Attention Network for Referring Expression Comprehension. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1307–1315.
15. Hu, R.; Rohrbach, M.; Andreas, J.; Darrell, T.; Saenko, K. Modeling Relationships in Referential Expressions with Compositional Modular Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1115–1124.
16. Huang, S.; Chen, Y.; Jia, J.; Wang, L. Multi-View Transformer for 3D Visual Grounding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 15524–15533.
17. Yang, L.; Xu, Y.; Yuan, C.; Liu, W.; Li, B.; Hu, W. Improving Visual Grounding with Visual-Linguistic Verification and Iterative Reasoning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 9499–9508.
18. Dai, B.; Zhang, Y.; Lin, D. Detecting Visual Relationships with Deep Relational Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3076–3086.
19. Lu, C.; Krishna, R.; Bernstein, M.S.; Li, F. Visual Relationship Detection with Language Priors. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 852–869.
20. Xu, D.; Zhu, Y.; Choy, C.B.; Fei-Fei, L. Scene Graph Generation by Iterative Message Passing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5410–5419.
21. Li, Y.; Ouyang, W.; Zhou, B.; Wang, K.; Wang, X. Scene Graph Generation from Objects, Phrases and Region Captions. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1261–1270.
22. Zellers, R.; Yatskar, M.; Thomson, S.; Choi, Y. Neural Motifs: Scene Graph Parsing with Global Context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5831–5840.
23. Teng, Y.; Wang, L. Structured sparse r-cnn for direct scene graph generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 19437–19446.
24. Gao, K.; Chen, L.; Niu, Y.; Shao, J.; Xiao, J. Classification-then-grounding: Reformulating video scene graphs as temporal bipartite graphs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 19497–19506.
25. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Volume 28.
26. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
27. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Region-based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 142–158. [CrossRef] [PubMed]
28. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
29. Chen, S.; Zhao, Q. REX: Reasoning-aware and Grounded Explanation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 15586–15595.
30. Li, G.; Wei, Y.; Tian, Y.; Xu, C.; Wen, J.R.; Hu, D. Learning to Answer Questions in Dynamic Audio-Visual Scenarios. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 19108–19118.
31. Cirik, V.; Berg-Kirkpatrick, T.; Morency, L.P. Using Syntax to Ground Referring Expressions in Natural Images. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*, 6756–6764. [CrossRef]
32. Lyu, F.; Feng, W.; Wang, S. vtGraphNet: Learning weakly-supervised scene graph for complex visual grounding. *Neurocomputing* **2020**, *413*, 51–60. [CrossRef]
33. Hinami, R.; Matsui, Y.; Satoh, S. Region-based Image Retrieval Revisited. In Proceedings of the ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 528–536.
34. Zhang, H.; Niu, Y.; Chang, S.F. Grounding referring expressions in images by variational context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4158–4166.

35. Yan, Y.; Ren, J.; Tschannerl, J.; Zhao, H.; Harrison, B.; Jack, F. Nondestructive phenolic compounds measurement and origin discrimination of peated barley malt using near-infrared hyperspectral imagery and machine learning. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 5010715. [CrossRef]
36. Sun, H.; Ren, J.; Zhao, H.; Yuen, P.; Tschannerl, J. Novel gumbel-softmax trick enabled concrete autoencoder with entropy constraints for unsupervised hyperspectral band selection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5506413. [CrossRef]
37. Sun, G.; Zhang, X.; Jia, X.; Ren, J.; Zhang, A.; Yao, Y.; Zhao, H. Deep fusion of localized spectral features and multi-scale spatial features for effective classification of hyperspectral images. *Int. J. Appl. Earth Obs. Geoinf.* **2020**, *91*, 102157. [CrossRef]
38. Qiao, T.; Ren, J.; Wang, Z.; Zabalza, J.; Sun, M.; Zhao, H.; Li, S.; Benediktsson, J.A.; Dai, Q.; Marshall, S. Effective denoising and classification of hyperspectral images using curvelet transform and singular spectrum analysis. *IEEE Trans. Geosci. Remote Sens.* **2016**, *55*, 119–133. [CrossRef]
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
40. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet Classification with Deep Convolutional Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; Volume 25.
41. Lyu, F.; Wu, Q.; Hu, F.; Wu, Q.; Tan, M. Attend and Imagine: Multi-label Image Classification with Visual Attention and Recurrent Neural Networks. *IEEE Trans. Multimed.* **2019**, *21*, 1971–1981. [CrossRef]
42. Du, K.; Lyu, F.; Hu, F.; Li, L.; Feng, W.; Xu, F.; Fu, Q. AGCN: Augmented Graph Convolutional Network for Lifelong Multi-label Image Recognition. *arXiv* **2022**, arXiv:2203.05534.
43. Mukhiddinov, M.; Cho, J. Smart glass system using deep learning for the blind and visually impaired. *Electronics* **2021**, *10*, 2756. [CrossRef]
44. Zagoruyko, S.; Komodakis, N. Learning to Compare Image Patches via Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4353–4361.
45. Tseng, S.Y.R.; Chen, H.T.; Tai, S.H.; Liu, T.L. Non-local RoI for Cross-Object Perception. *arXiv* **2018**, arXiv:1811.10002.
46. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
47. Buades, A.; Coll, B.; Morel, J.M. A Non-local Algorithm for Image Denoising. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 60–65.
48. Ye, Z.; Hu, F.; Liu, Y.; Xia, Z.; Lyu, F.; Liu, P. Associating multi-scale receptive fields for fine-grained recognition. In Proceedings of the IEEE International Conference on Image Processing, Abu Dhabi, United Arab Emirates, 25–28 October 2020; pp. 1851–1855.
49. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated Residual Transformations for Deep Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.
50. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Hierarchical Question-Image Co-Attention for Visual Question Answering. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 1–9.
51. Wang, P.; Wu, Q.; Shen, C.; van den Hengel, A. The Vqa-Machine: Learning How to Use Existing Vision Algorithms to Answer New Questions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1173–1182.
52. Fukui, A.; Park, D.H.; Yang, D.; Rohrbach, A.; Darrell, T.; Rohrbach, M. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–4 November 2016; pp. 457–468.
53. Luo, R.; Shakhnarovich, G. Comprehension-Guided Referring Expressions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7102–7111.
54. Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; Darrell, T. Natural Language Object Retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4555–4564.
55. Mao, J.; Huang, J.; Toshev, A.; Camburu, O.; Yuille, A.L.; Murphy, K. Generation and Comprehension of Unambiguous Object Descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 11–20.
56. Yu, L.; Poirson, P.; Yang, S.; Berg, A.C.; Berg, T.L. Modeling Context in Referring Expressions. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 69–85.
57. Nagaraja, V.K.; Morariu, V.I.; Davis, L.S. Modeling Context between Objects for Referring Expression Understanding. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 792–807.
58. Kazemzadeh, S.; Ordonez, V.; Matten, M.; Berg, T.L. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014; pp. 787–798.
59. Lin, T.; Maire, M.; Belongie, S.J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 8–14 September 2014; pp. 740–755.
60. Deng, J.; Dong, W.; Socher, R.; Li, L.j.; Li, K.; Fei-fei, L. ImageNet : A Large-Scale Hierarchical Image Database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
61. Yu, L.; Tan, H.; Bansal, M.; Berg, T.L. A Joint Speaker-Listener-Reinforcer Model for Referring Expressions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7282–7290.