applied sciences

MDPI

Article

Object Detection Based on Multiple Information Fusion Net

Yanni Zhang [1], Jun Kong [1,2], Miao Qi [1], Yunpeng Liu [1], Jianzhong Wang [1,*] and Yinghua Lu [2,*]

[1] College of Information Science and Technology, Northeast Normal University, Changchun 130000, China; zhangyn500@nenu.edu.cn (Y.Z.); kongjun@nenu.edu.cn (J.K.); qim801@nenu.edu.cn (M.Q.); Liuyp437@nenu.edu.cn (Y.L.)

[2] Institute for Intelligent Elderlycare, College of Humanities and Sciences, Northeast Normal University, Changchun 130000, China

* Correspondence: wangjz019@nenu.edu.cn (J.W.); luyh@nenu.edu.cn (Y.L.)

Received: 23 November 2019; Accepted: 1 January 2020; Published: 6 January 2020

Abstract: Object detection has been playing a significant role in computer vision for a long time, but it is still full of challenges. In this paper, we propose a novel object detection framework based on relationship among different objects and the scene-level information of the whole image to cope with the problem that some strongly correlated objects are difficult to be recognized. Our motivation is to enrich the semantics of object detection feature by a scene-level information branch and a relationship branch. There are three important changes of our framework over traditional detection methods: representation of relationship, scene-level information as the prior knowledge and the fusion of the above two information. Extensive experiments are carried out on PASCAL VOC and MS COCO databases. The experimental results show that the detection performance can be improved by introducing relationship and scene-level information, and our proposed model achieve better performance than several classical and state-of-the-art methods.

Keywords: object relationship; scene-level information; information fusion; object detection

1. Introduction

Object detection is a hot topic in the field of computer vision and machine learning due to their widely applications in autonomous driving, robots, video surveillance, pedestrian detection, and so on. The classical object detection techniques are mainly based on the use of manual features, which can be divided into three steps: (1) target area selection; (2) feature extraction; (3) classification. In the first step, sliding-window strategy [1] which utilizes the sliding-windows with different dimensions and length-width ratios is widely adopted to search for candidate regions exhaustively. In the second step, the candidate regions obtained in the first step are analyzed. Several techniques can be used in this step for feature extraction, such as scale-invariant feature transform (SIFT) [2], histogram of oriented gradients (HOG) [3] and speeded-up robust features (SURF) [4]. In the third step, the candidate regions are classified according to the features extracted in the previous step by using classifiers such as support vector machine (SVM) [5] and AdaBoost [6]. Although the classical methods have been adopted in some object detection problems, there are still some limitations that hinder their breakthrough in speed and accuracy. Firstly, since the sliding-window strategy will capture many candidate regions in the original image, and the feature of regions needs to be extracted one by one, the classical object detection approaches are time-consuming. Secondly, the classical object detection methods may lack robustness because artificially designed features are sensitive to the variance in morphology, illumination and occlusion of object.

Recently, some deep learning techniques have been applied to object detection to overcome the limitations of traditional approaches [7–13]. The current state-of-the-art detector based on the deep learning can be roughly divided into two categories. One is two-stage methods which first form a series of candidate object proposals by Selective Search [8], EdgeBoxes [9], DeepMask [12] or region proposal network (RPN) [7], and then input the proposals into convolutional neural network for classification. The other is one-stage methods which straightforwardly predict confidences and locations of multiple objects on the whole feature map without generating candidate object proposals.

Region-based convolutional network (R-CNN) [14], as the beginning of combining object detection and deep learning, is a representative two-stage based approach. It achieves excellent object detection accuracy by extracting CNN features from the candidate regions and applies linear SVMs as the classifier. However, since the ConvNet forward pass is performed for each object proposal independently, the computational cost of R-CNN is high. Furthermore, the multi-stage training strategy which contains feature extraction, fine-tuning network, training SVMs, and bounding-box regression also makes the training of R-CNN be slow. In [15], a spatial pyramid pooling network (SPPNet) was proposed. Although SPPNet can speed up R-CNN by sharing computation, its training is also a multi-stage pipeline. Besides, the fine-tuning algorithm proposed in SPPNet cannot update the convolutional layer, which limits its accuracy when the networks are very deep. For the sake of further decreasing the computational cost and improving the accuracy of object detection, Ross et al. proposed a fast region-based convolutional network (Fast R-CNN) [16]. The Fast R-CNN utilizes a novel RoI-pooling operation to extract feature vectors for each candidate region from shared convolutional feature map, which greatly improves the processing speed. In Fast R-CNN, the detection accuracy can also be enhanced by updating all network layers during training. Although SPPNet and Fast R-CNN have effectively reduced the training time of object detection networks, the region proposal computation is still considered as a bottleneck in them. To deal with this issue, Ren et al. proposed a Faster R-CNN [7] which replaces the Selective Search method with RPN to achieve end-to-end training. RPN is a kind of fully convolutional network (FCN) [17]. By sharing full-image convolutional features with the detection network, RPN enables nearly cost-free region proposals to solve the time-consuming problem of Fast R-CNN. However, the multiple scale proposals generated by sliding a fixed set of filters over a fixed set of convolutional feature maps in RPN may be inconsistent with the sizes of objects. Thus, Cai et al. proposed a multi-scale CNN (MS-CNN) [18] to match the receptive fields to different scales of objects and employed a multiple output layer for object detection. Recently, for the purpose of improving the detection performance, some more state-of-the-art techniques (such as Resnet [19] and Inception series [20–22]) were employed to replace the standard CNN as the backbone networks of the two-stage based object detection methods, which can be found in the object detection API from google [23].

Different from the aforementioned methods, the one-stage approaches can achieve complete single network training under the premise of guaranteeing a certain accuracy rate. You only look once (YOLO) [24], YOLO9000 [25], an iterative grid based object detector (G-CNN) [26], and single shot multibox detector) [27] are representative techniques in this category. Through treating the object detection task as a regression problem, YOLO spatially separates bounding boxes and associated class probabilities. Since the whole detection pipeline of YOLO is a single network, an end-to-end optimization of the network can be directly performed. SSD combines predictions of multiple feature maps with different resolutions to detect objects of various sizes. Since the proposal generation, subsequent pixel and feature resampling stages are eliminated in SSD, it can be easily trained. Although the running speed of one-stage methods can be significantly improved, their accuracy is always inferior to the two-stage approaches [27]. To address this issue, the Resnet and Inception have also been utilized [23]. Furthermore, Lin et al. replaced the standard cross entropy loss with a novel Focal loss and proposed a RetinaNet to solve the class imbalance problem in one-stage based object detection [28].

No matter the approach belongs to one-stage or two-stage, most the aforementioned algorithms do not effectively utilize the relationship among objects, but only use the feature associated with the

object itself for detection. Recently, some researchers have realized the importance of relation, and proposed some methods [29–31] to achieve better detection results by exploring the relationships between objects. In ION [32], Bell et al. proposed a spatial recurrent neural networks (RNNs) for exploring contextual information across the entire image. Xu et al. put forward a scene graph generation approach by iterative message passing [33]. The network regards a single object as a point in topology, and the relationships of objects are considered as edges connecting points. Through passing information between the edges and points, it is proved that the relationship between objects has a positive impact on detection. Georgia et al. proposed a human-centric based model called InteractNet [34], in which human is regarded as the main clue to establish a relationship with other surrounding objects. The InteractNet indicates that a person's external behavior can provide powerful information to locate the objects they are interacting with. Liu et al. proposed a structure inference net (SIN) [35] which explores the structure relationship between objects for detection. However, SIN only takes the spatial coordinates of object proposals into account, while the appearance feature of object proposals is neglect. Han et al. presented a relation network [36], which considers both the appearance and geometry feature of object proposals for relation construction. Nevertheless, the scene-level feature which could provide a lot of context information for object detection [37] is ignored in relation Nntwork.

This paper proposes a novel object detection algorithm based on multiple information fusion net (MIFNet). Compared with the existing techniques, our algorithm not only adaptively establishes relationships between objects through attention mechanism [38], but also introduces scene-level information to make the proposed approach richer in semantics. In MIFNet, the relationships between an object and all other objects are got by relation channel modules. Besides, by introducing the scene-level context [21,39,40], the proposed network can enrich the object feature with scene information. The experimental results on PASCAL VOC [41] and MS COCO [42] databases demonstrate the effectiveness of the proposed algorithm.

The paper is structured as follows. The related work is introduced in Section 2. The proposed MIFNet is described in Section 3. The experimental results are given in Section 4. The conclusion is provided in Section 5.

2. Related Work

Context information: In real life, it is unlikely that an object can exist alone. Visual objects occur in particular environments and usually coexist with other related objects [43]. When the object's appearance feature is insufficient because of small object size, object occlusion, or poor image quality, a proper modeling of context will facilitate object detection and recognition task. Context information has been applied in many methods to enhance the performance of object detection [44–49], which can be roughly divided into two categories [49,50]: global information [32,51] (refers to the image level or scene level information), local information [35,36] (considers the object relationship or the interaction between the object and its surrounding area). It is proved that both the global and local context information have a positive impact on the object detection. Our proposed MIFNet has the capability of utilizing both global context (scene-level information) and local information (object relationship) to make the object's appearance feature richer.

Attention mechanism: The attention mechanism in deep learning is inspired by the mode of human attention thinking and has been widely used in natural language processing [52]. In attention module, an individual element can be influenced by aggregating information from other elements and the dependency between elements is modeled without excessive assumptions on their locations and feature distributions. The aggregation weights can be learned automatically, which is driven by the task goal. Recently, attention mechanism has been successfully applied in vision problems [37,53].

Attention mechanism can be represented as follows:

$$\text{Attention}\ (Q,\ K,\ V)\ =\ \text{softmax}\left(\frac{QK^T}{\sqrt{d}}V\right), \tag{1}$$

where Q, K, V are three feature matrices with the same shape. In natural language processing, Q, K, V represent sets of queries, keys, and values, respectively. In our work, Q denotes the object feature, K denotes all surrounding object feature, V represents the all image feature (with object feature and location feature). In Equation (1), given a query $q \in Q$ and all keys K, the dot products of q with K will be calculated to get the similarity between them. Then we divide dot products by a scaling factor $\sqrt{d}$ and the softmax function is applied to obtain the weights on all image feature (i.e., it can obtain the influence of each object on the current object). For more detailed information about the attention mechanism, the readers can refer to [38]. In our work, the attention mechanism is utilized to get the relationship between objects.

3. The Proposed Method

The framework of proposed multiple information fusion net (MIFNet) is shown in Figure 1. In our MIFNet, the feature map of an input image is first obtained through a feedforward convolutional network (VGG or Resnet). In the next stage, the network feature map is divided into two parts. One is as a part of the input of the first branch and the other is utilized to get the region proposals through RPN and then served as the input of the second branch. In the first branch network (I), a series of operations is performed on the feature map of the entire image to get the scene-level information as the input of scene GRU (Gated Recurrent Unit in III up). In the second branch network (II), the attention mechanism is utilized to establish object relationships adaptively. For the purpose of classifying and regressing regions of interest (RoIs), the second branch network not only utilizes the appearance feature extracted by convolutional layers and the coordinate information of the object, but also the information of all surrounding objects as the input of relation GRU (in III below). In the message passing module (III), scene GRUs and relation GRUs communicate information to each other in order to keep up with new information. In the last stage, we concatenate the information obtained by these two GRUs to refine the position of the corresponding RoI and predict the category of objects.

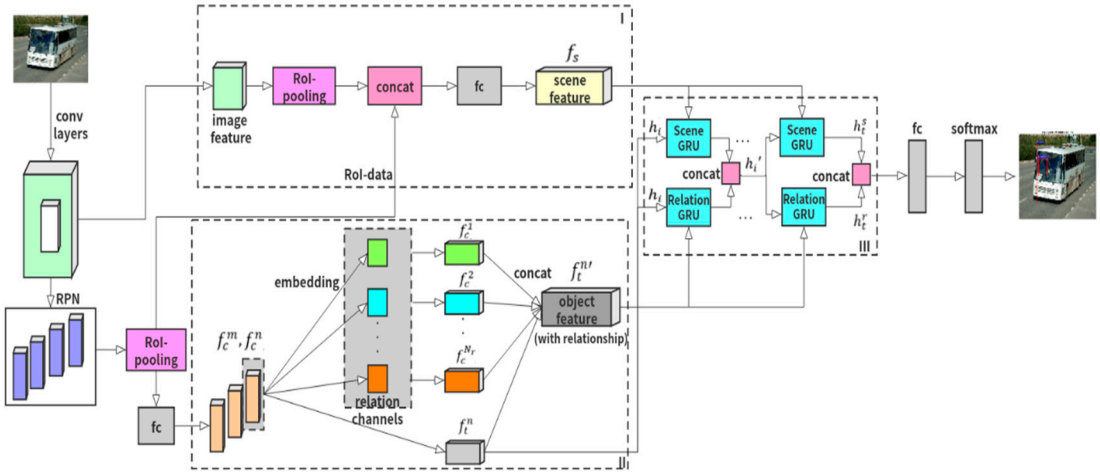


Figure 1. The framework of our approach. I: Scene-level information processing module. II: Relationship module. III: Message passing module.

3.1. Scene-Level Information Processing Module

Contextual information is important for accurate object recognition. To extract the scene-level information, the image feature is firstly obtained by convolutional network (VGG or Resnet) as the input of the first branch. Secondly, the image feature obtained by RoI-pooling layer and the feature obtained by RPN (without scene information) are concatenated as the input of a convolutional layer. By concatenation, the information of potential object is richer. Besides, the weight of potential object

can also be increased by training. In the end, the output f_s of the first network branch, which called scene feature, is input to the scene GRU to choose information and update object feature.

3.2. Relationship Module

In most previous object detection methods based on the convolutional neural network [7,16], each object is identified independently, and the relationship of objects is neglected. To overcome this limitation, the proposed approach models the relationship of objects by groups. That is, the feature vector of an object is obtained by fusing the features of itself and other objects to enrich the information, as shown in Figure 2.

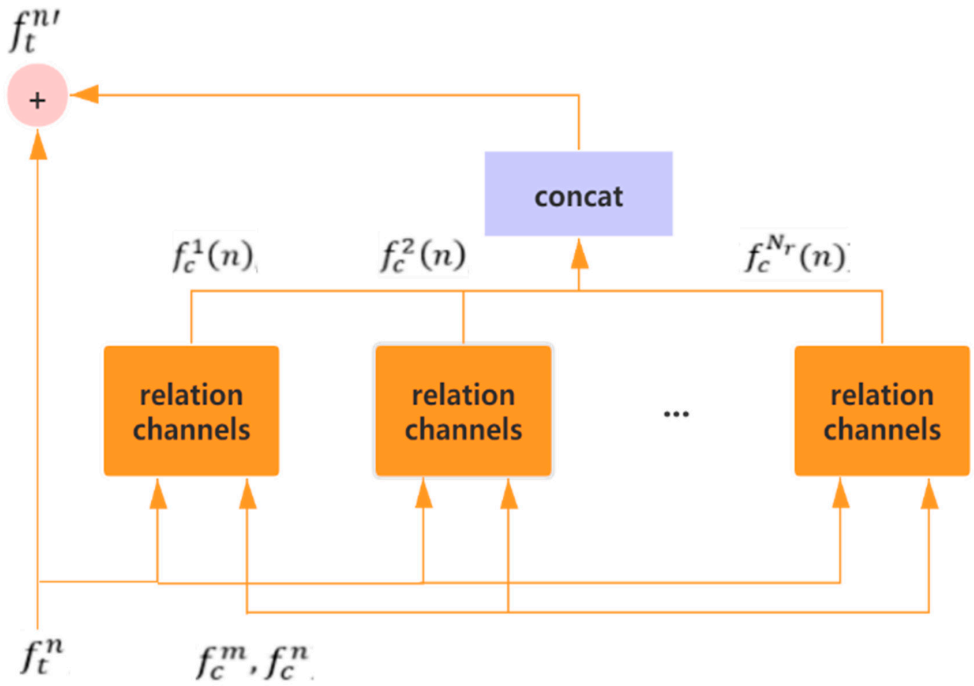


Figure 2. Object relation module.

In Figure 2, given an input set of N objects $\left\{\left(f_t^n,\ f_c^n\right)\right\}_{n=1}^{N}$, in which f_t^n is the original appearance feature of the nth object extracted by convolutional neural network, f_c^n denotes the location feature of the nth object composed by the 4-dimensional feature of object bounding box. The bounding box feature comprises width (w), height (h) and the center coordinates (x, y) of the box in our study. The relationship channel is a module that handles relationships among different objects, N_r is the number of channels ($N_r = 64$). By object relation module, the $f_c^1(n),\ f_c^2(n) \ldots f_c^{N_r}(n)$ which fuse the location information of all surrounding objects can be gained. For the purpose of obtaining the output $f_t^{n'}$ that is finally sent into the relation GRU, we concatenate the vectors on all channels $f_c^1(n), f_c^2(n) \ldots f_c^{N_r}(n)$. Because the processing mechanisms of relation channel modules are the same, we take one relation channel module as an example to explain how relation channel works.

Figure 3 shows the process of one relation channel module. Firstly, the dot product operation is applied to obtain the appearance weight w_t^{mn} between the mth and nth objects, as shown in Equation (2).

$$w_t^{mn} = \frac{(W_K f_t^m)\cdot(W_Q f_t^n)}{\sqrt{d}}, \tag{2}$$

where W_K, W_Q are matrices which map the original appearance f_t^m and f_t^n into subspaces, $\cdot$ denotes the operation of dot product to obtain the degree of matching between $W_K f_t^m$ and $W_Q f_t^n$.

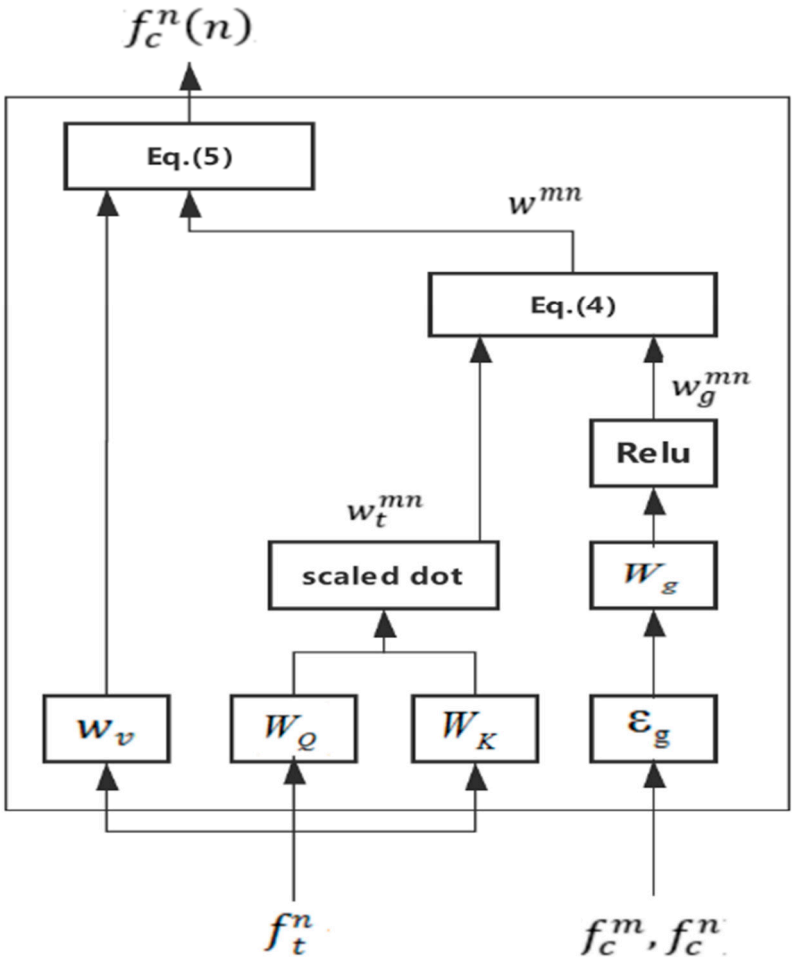


Figure 3. Relation channel module.

Secondly, the location weight w_g^{mn} is calculated by Equation (3).

$$w_g^{mn} = \text{Relu}\left(W_g \cdot \varepsilon_g\left(f_c^m, f_c^n\right)\right), \tag{3}$$

where f_c^m and f_c^n are geometry features that contain six relative position information ($\log\left(\frac{|x_m - x_n|}{w_n}\right)$, $\log\left(\frac{|y_m - y_n|}{h_n}\right)$, $\log\left(\frac{w_m}{w_n}\right)$, $\log\left(\frac{h_m}{h_n}\right)$, $\sqrt{\frac{x_m - x_n}{w_n}}$, $\sqrt{\frac{y_m - y_n}{h_n}}$), in which w_n, h_n, x_n and y_n are the width, height and center coordinates of the nth object, ε_g is a function based on sine and cosine to embed the geometry features into high-dimensional space [38]. Then we use W_g to convert the embedded vector to scalar weight. The Relu activation function is utilized to ensure that only objects with certain geometric relationship can participate in this relationship calculation.

Next, the relationship weight is obtained by Equation (4).

$$w^{mn} = \text{softmax}\left(w_g^{mn} \cdot \exp\left(w_t^{mn}\right)\right), \tag{4}$$

where the relationship weight w^{mn} represents the impact of the mth object to the nth object, softmax is employed for normalization. Finally, Equation (5) can be utilized to get a feature $f_c^{N_r}(n)$ that has the influence of surrounding objects on it.

$$f_c^{N_r}(n) = \sum_m w^{mn} \cdot \left(W_v f_t^n\right), \tag{5}$$

where W_v is used to transform the original appearance feature f_t^n linearly. Equation (5) is the process of integrating the information of the object and other objects into the original appearance feature. The output $f_c^{N_r}(n)$ is the weighted sum of the initial appearance features from other objects, which contains both its original appearance feature and the feature of all objects around it.

In the end, by the relation channel module, the feature $f_t^{n'}$ which merges features of multiple channels can be gained by Equation (6).

$$f_t^{n'} = f_t^n + [f_c^1(n), \ldots, f_c^{N_r}(n)], \textit{for all } n. \tag{6}$$

where the fusion feature $f_t^{n'}$ includes the extracted original appearance feature f_t^n (the initial appearance feature after convolutional layers) and the relationship feature $(f_c^1(n), \ldots, f_c^{N_r}(n))$ (fusing the location information of all surrounding objects under a particular channel). In the relation channel, the feature of other objects can be mixed together to identify the relationship between the current object and other objects, and finally merged with the original appearance feature through the fully connected network. The final output $f_t^{n'}$ is the input of the Relation GRU.

3.3. Message Passing Module

As we have discussed previously, context information is important for accurate object detection. For example, in Figure 4a, if road is considered as the global or scene-level information, the objects in this image are hardly to be detected as ships and planes since it is generally impossible for them to appear in the road scene. Similarly, in Figure 4b, when a dinner table appears, the probability of detecting chairs increases, because the dinner tables and chairs always appear in pairs. Thus, the Gated Recurrent Unit (GRU) [54] is utilized in this study. Similar to the long short-term memory (LSTM) model [55], GRU unit also has the function of adjusting the information flow in the unit, but it is lightweight and effective [35]. In the message passing module, information is continuously passed between the scene GRU and the relation GRU so that the useful information can be preserved.

(**a**) (**b**)

Figure 4. Some detection results of our model. (**a**) is a picture of road scene, (**b**) is a picture of dining room.

GRU only has two gates. It combines the input gate and the forget gate in the LSTM into one, and the combined gate is called update gate, which determines how much information from the previous time and the current time is to be passed on. The other gate in GRU is reset gate, which controls how much past information is forgotten.

In our work, two parallel GRUs are applied to pass information to each other, one is the scene GRU and the other is the relation GRU. The scene GRU receives the whole image information f_s as the input. The input of relation GRU is the integrated object information $f_t^{n'}$, which includes the object's own information and the influence of surrounding objects on it. We represent the initial state h_i of the

network with the original appearance feature (without any scene information or relation information). Here, since the processing mechanisms of scene GRU and relation GRU are identical, we take the relation GRU as an example to show how GRU works. Firstly, the reset gate of the tth moment r_t is calculated as follows:

$$r_t = \sigma\left(W_r\left[f_t^{n'}, h_i\right]\right), \tag{7}$$

where σ is the logistic sigmoid function, [,] denotes the concatenation of vectors, and W_r is a weight matrix learned through the convolutional neural network. The output of reset gate r_t determines whether the previous state is forgotten. When r_t is close to zero, the status information h_i of the previous moment will be forgotten, and the hidden status is reset to the current input. Similarly, the update gate of the tth moment z_t is computed by

$$z_t = \sigma\left(W_z\left[f_t^{n'}, h_i\right]\right), \tag{8}$$

where z_t is used to determine how much past information can continue to be passed on, W_z is a weight matrix. If the value of the update gate is larger, the state information introduced at the previous moment is more, and vice versa. In GRU, the new hidden state $\widetilde{h_t}$ can be obtained through Equation (9):

$$\widetilde{h_t} = tanh(W\left[r_t * h_i, f_t^{n'}\right]), \tag{9}$$

where the new hidden state $\widetilde{h_t}$ is determined by the value of the reset gate, W is a weight matrix, $*$ denotes the element-wise multiplication. The actual output h_{i+1} is then computed by

$$h_{i+1} = 1 - z_t * h_i + z_t * \widetilde{h_t}, \tag{10}$$

where some of the previous state h_i will be passed, and the new hidden state $\widetilde{h_t}$ will be selectively updated. Through GRU, the scene module and the relation module can pass information to each other and constantly update new information. In this way, useful information will continue to be passed, and useless information will be ignored. Finally, richer information can be obtained through Equation (11).

$$h_i' = \frac{\left[h_{i+1}^s, h_{i+1}^r\right]}{2}, \tag{11}$$

where h_{i+1}^s represents the information obtained by the scene GRU, and h_{i+1}^r denotes the information obtained by the relation GRU. The integrated information h_i' will be sent into the next GRUs as the new initial state.

4. Experimental Results

4.1. Experimental Settings

Databases and evaluation metrics: Our model was evaluated on two databases: PASCAL VOC [41] and MS COCO [42]. PASCAL VOC is a widely used image database for object detection and classification. In our work, VOC 2007 and VOC 2012 subsets were utilized. VOC 2007 data set contained 9963 annotated images and 24,640 annotated objects, which were composed of three parts: train, validation (val), and test sets. VOC 2012 is an updated version of VOC 2007 data set, which included 11,530 pictures with 20 categories including people, animals (such as cats, dogs and birds), vehicles (such as cars, ships and planes), and furniture (such as chairs, tables, and sofas). Some examples of PASCAL VOC database can be seen in Figure 5. MS COCO database is built by Microsoft, which contained 328,000 images with 2000 object labels. Compared with PASCAL VOC, MS COCO includes natural images and common object images in daily life with more complex background, larger number of targets and smaller size. Thus, the object detection task on MS COCO database was more difficult and challenging. The sample images in MS COCO are shown in Figure 6. In this

study, we followed Ref. [35] to adopt average precision (AP) and mean of average precision (mAP) as our evaluation metrics to compare the performances of different approaches. AP, which derived from precision and recall, is one of the popular metrics to measure the accuracy of object detectors. It computes the average precision value for recall value over 0 to 1. The precision and the recall rates are defined as follows:

$$precision = \frac{TP}{TP + FP}, \tag{12}$$

$$recall = \frac{TP}{TP + FN}, \tag{13}$$

where TP represents the number of true positive examples, FP represents the number of fault positive examples, and FN represents the number of fault negative examples. mAP is the average of AP on all categories.

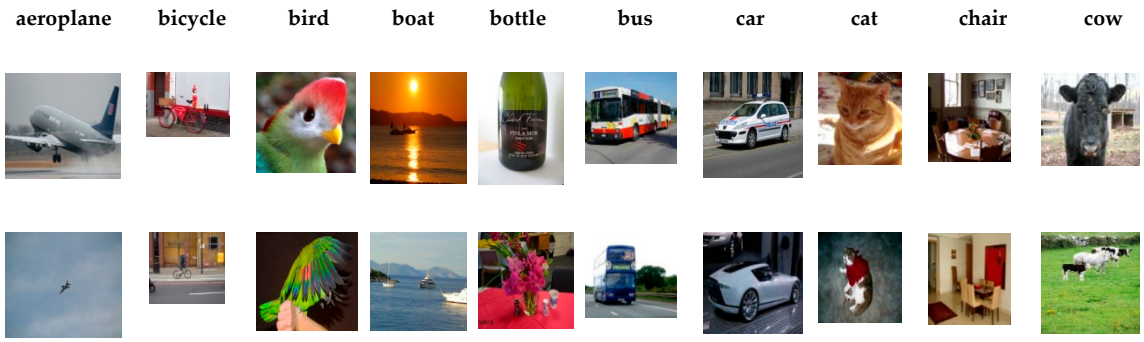


Figure 5. Some sample images in PASCAL VOC database.

Figure 6. Some sample images in MS COCO database.

Implementation details: In our experiments, the proposed model was implemented based on Faster R-CNN [7], an open-source framework for object detection built on Tensorflow [56] platform. The VGG-16 [57] and Resnet-101 [19] pre-trained on ImageNet [58] were adopted as backbone networks in our model to extract image feature. When adding the newly fully connected and convolutional layers, they were randomly initialized with a zero-mean Gaussian distribution with standard deviations of 0.01 and 0.001. The message passing module contained two parallel GRU units with shared parameters. All the parameters of the GRU units were initialized based on SIN [35]. Non-maximum suppression (NMS) with intersection over union (IOU) were used for duplicate removal in all experiments.

Stochastic gradient descent (SGD) was applied to fine tune our network. Each SGD mini batch was composed of 256 randomly sampled object proposals from two randomly chosen images. In each mini-batch, 25% of the RoIs were selected as foreground from object proposals, which had IOU overlapped with a ground-truth bounding box of at least 0.5. We sampled the remaining RoIs from object proposals which had a maximum IOU with ground truth in the interval [0:1; 0:5]. We trained

our model on a single NVIDIA GeForce GTX TITAN X GPU with 12 GB memory. The experimental parameters, training and test time of our MIFNet can be seen in Table 1.

Table 1. Experimental parameter setting.

Train Database	Test Database	Backbone	Iterations Setting	Learning Rate	Run Time (Sec/Img)	Test Time (Sec/Img)
VOC 2007 trainval + VOC 2012 trainval	VOC 2007 test	VGG16	first 80k iterations last 50k iterations	0.0005 0.00005	0.35	0.17
		Resnet-101	first 80k iterations last 30k iterations	0.001 0.0001	0.46	0.19
VOC 2007 trainval + VOC 2012 trainval + VOC 2007 test	VOC 2012 test	VGG16	first 100k iterations last 70k iterations	0.0005 0.00005	0.35	0.15
		Resnet-101	first 100k iterations last 50k iterations	0.001 0.0001	0.46	0.18
MS COCO 2014 train + MS COCO 2014 val	MS COCO 2014 minival	VGG16	first 350k iterations last 200k iterations	0.0005 0.00005	0.32	0.16
		Resnet-101	first 350k iterations last 150k iterations	0.001 0.0001	0.46	0.19

4.2. Performance Comparisons

PASCAL VOC Database: The performance of our MIFNet was compared with some classical and state-of-the-art object detection methods, including Fast R-CNN [16], Faster R-CNN [7], SIN [35], ION [32], CPF [40], and so on. The experimental results on data set VOC 2007 test and VOC 2012 test are shown in Tables 2 and 3 respectively. All the experimental results of comparison approaches are quoted from their corresponding literatures. From these tables, the following points can be observed. Firstly, Fast R-CNN and Faster R-CNN are classical two-stage approaches, while SSD is a classical one-stage approach. Since the relationship between objects and context information were neglected in them, their performances were inferior to other approaches. Secondly, by utilizing the spatial recurrent neural networks and semantic segmentation, ION and CPF took global contextual information into account. Therefore, they outperformed the classical approaches such as Fast R-CNN, Faster R-CNN, and SSD. Thirdly, SIN considered both the scene context information and object relationships. However, the relationship in SIN was established only by geometric structure of the objects, which neglected the objects' appearance information. As a result, its performance was still worse than the proposed MIFNet. At last, the proposed approach leveraged the attention mechanism to adaptively establish the relationship between objects, which considered both geometric and appearance information. Besides, the scene-level information was also introduced into the model. Thus, our MIFNet greatly improved the detection accuracy of some small and highly correlated objects (such as chair, boat, plant, tv) and achieved the best performance on PASCAL VOC database.

Table 2. Detection results on PASCAL VOC 2007 test. Train set: 07 trainval + 12 trainval. "V" and "R" denote the model uses VGG-16 and Resnet-101 as backbone networks, respectively. The bold characters represent the best result for each column.

Method	Net	mAP	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	PLANT	Sheep	Sofa	Train	Tv
Fast R-CNN [16]	V	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
Faster R-CNN [7]	V	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
SSD500 [24]	V	75.1	79.8	79.5	74.5	63.4	51.9	84.9	85.6	87.2	56.6	80.1	70.0	85.4	84.9	80.9	78.2	49.0	78.4	72.4	84.6	75.5
ION [32]	V	75.6	79.2	83.1	77.6	65.6	54.9	85.4	85.1	87.0	54.4	80.6	73.8	85.3	82.2	82.2	74.4	47.1	75.8	72.7	84.2	80.4
SIN [35]	V	76.0	77.5	80.1	75.0	67.1	62.2	83.2	86.9	88.6	57.7	84.5	70.5	86.6	85.6	77.7	78.3	46.6	77.6	74.7	82.3	77.1
Faster R-CNN [19]	R	76.4	79.8	80.7	76.2	68.3	55.9	85.1	85.3	**89.8**	56.7	**87.8**	69.4	88.3	**88.9**	80.9	78.4	41.7	78.6	79.8	85.3	72.0
SSD321 [24]	R	77.1	76.3	84.6	79.3	64.6	47.2	85.4	84.0	88.8	60.1	82.6	**76.9**	86.7	87.2	**85.4**	79.1	50.8	77.2	**82.6**	**87.3**	76.6
MIFNet(ours)	V	77.6	79.2	79.8	77.4	71.4	63.3	86.3	87.1	89.4	63.1	85.1	72.4	86.7	86.8	78.2	79.0	50.7	77.9	75.4	84.8	77.6
MIFNet(ours)	R	**80.6**	**81.6**	**86.4**	**80.9**	**72.6**	**70.2**	**87.7**	**88.5**	88.5	**66.8**	87.1	73.8	**89.0**	87.5	83.8	**82.5**	**55.2**	**83.1**	79.6	85.5	**81.6**

Table 3. Detection results on PASCAL VOC 2012 test. Train set: 07trainval + 12trainval + 07test. "V" and "R" denote the model uses VGG-16 and Resnet-101 as backbone networks, respectively. The bold characters represent the best result for each column.

Method	Net	Map	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	Tv
Fast R-CNN [16]	V	68.4	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73.0	55.0	87.5	80.5	80.8	72.0	35.1	68.3	65.7	80.4	64.2
SSD300 [24]	V	70.3	84.2	76.3	69.6	53.2	40.8	78.5	73.6	88.0	50.5	73.5	61.7	85.8	80.6	81.2	77.5	44.3	73.2	66.7	81.1	65.8
Faster R-CNN [7]	V	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
CPF [40]	V	72.6	84.0	81.2	75.9	60.4	51.8	81.2	77.4	90.9	50.2	77.6	58.7	88.4	83.6	82.0	80.4	41.5	75.0	64.2	82.9	65.1
SIN [35]	V	73.1	84.8	79.5	74.5	59.7	55.7	79.5	78.8	89.9	51.9	76.8	58.2	87.8	82.9	81.8	81.6	51.2	75.2	63.9	81.8	67.8
SSD321 [24]	R	75.4	87.9	82.9	73.7	61.5	45.3	81.4	75.6	92.6	57.4	78.3	**65.0**	90.8	86.8	**85.8**	81.5	50.3	78.1	**75.3**	85.2	72.5
R-FCN [17]	R	77.6	86.9	83.4	81.5	63.8	**62.4**	81.6	81.1	93.1	58.0	**83.8**	60.8	**92.7**	86.0	84.6	84.4	**59.0**	80.8	68.6	86.1	72.9
MIFNet(ours)	V	74.4	86.2	81.9	76.4	60.3	58.0	80.2	78.4	90.4	53.8	78.5	58.3	88.3	83.0	83.7	82.7	53.0	76.3	66.5	82.6	70.5
MIFNet(ours)	R	**78.4**	**87.7**	**84.3**	**82.8**	**66.5**	60.4	**86.0**	**82.2**	**93.7**	**59.0**	76.5	62.2	91.3	**87.6**	84.8	**85.9**	58.3	**81.3**	72.1	**86.4**	**79.2**

MS COCO database: In order to further verify the effectiveness of our proposed method, MS COCO database was utilized. The object detection results of different methods on this database are tabulated in Table 4. In this table, AP was averaged precision across all object categories and multiple intersection over union (IOU) values from 0.5 to 0.95, AP^{50} denotes the mAP at IOU = 0.50, AP^{70} denotes the mAP at IOU = 0.70, average recall (AR) represents the average recall rate averaged over all categories and IOU thresholds. AR^1, AR^{10} and AR^{100} denote the maximum recall rate of the fixed number (1, 10, 100) of objects detected in each image, AR^S, AR^M and AR^L represent the recall rate of small (area smaller than 32^2), medium (area between 32^2 and 96^2) and large (area bigger than 96^2) objects respectively. From Table 4, we can get the following observations. Firstly, since ION takes global context information into consideration, its performance is better than the classical approaches such as Fast R-CNN, Faster R-CNN and YOLOv2. Secondly, SIN outperforms ION, which indicates the object relationship is important for object detection. Finally, the proposed MIFNet performs best on MS COCO database because it establishes the relationship between objects by both geometric and appearance information adaptively and takes the scene-level information into account. In summary, these observations are consistent with the experimental results of PASCAL VOC database.

Table 4. Detection results on MS COCO 2014 minival. Train set: trainval35k: MS COCO train + 35k val. "V" and "R" denote the model uses VGG-16 and Resnet-101 as backbone networks, respectively. The bold characters represent the best result for each column.

Method	Net	AP	AP^{50}	AP^{70}	AP^{S}	AP^{M}	AP^{L}	AR^{1}	AR^{10}	AR^{100}	AR^{S}	AR^{M}	AR^{L}
Fast R-CNN [16]	V	20.5	39.9	19.4	4.1	20.0	35.8	21.3	29.5	30.1	7.3	32.1	52.0
Faster R-CNN [7]	V	21.1	40.9	19.9	6.7	22.5	32.3	21.5	30.4	30.8	9.9	33.4	49.4
YOLOv2 [27]	V	21.6	44.0	19.2	5.0	22.4	35.5	20.7	31.6	33.3	9.8	36.5	54.4
ION [32]	V	23.0	42.0	23.0	6.0	23.8	37.3	23.0	32.4	33.0	9.7	37.0	53.5
SIN [35]	V	23.2	44.5	22.0	7.3	24.5	36.3	22.6	31.6	32.0	10.5	34.7	51.3
SSD321 [24]	R	28.0	45.4	29.3	6.2	28.3	49.3	25.9	37.8	39.9	11.5	43.3	64.9
DSSD321 [24]	R	28.0	46.1	29.2	7.4	28.1	47.6	-	-	-	-	-	-
R-FCN [17]	R	29.9	51.9	-	10.8	32.8	45.0	-	-	-	-	-	-
SSD513 [24]	R	31.2	50.4	33.3	10.2	34.5	**49.8**	28.3	42.1	44.4	17.6	**49.2**	**65.8**
MIFNet(ours)	V	29.3	53.6	29.2	10.3	34.7	46.6	26.3	36.9	37.5	13.5	43.7	59.3
MIFNet(ours)	R	**32.1**	**55.3**	**33.5**	**15.8**	**36.8**	47.7	**28.4**	**42.4**	**43.4**	**24.2**	49.0	61.6

Effectiveness of scene-level information: For the purpose of verifying the effectiveness of each part in our proposed method, some ablation experiments are carried out. Here, we employ the VGG-16 as the backbone of our MIFNet. In the first experiment, only the scene-level information is considered to update the object feature. As shown in Tables 5 and 6, applying scene-level information achieves a better mAP of 75.8% compared with the baseline (Faster R-CNN without scene-level information and object relationships) on PASCAL VOC 2007. On the bigger database MS COCO, a better mAP of 23.5% can be obtained. We find that introducing the scene-level information can improve the detection performances in certain categories, including bike, bottle, chair, plant, TV, and so on. Especially, the average accuracy of plant has increased by more than 10%. These results are not surprising since these categories are usually highly relevant to the context of scene. From Figure 7, we can clearly see that the detection result in (b) with scene-level information is more accurate than the detection result in (a) without scene-information. This may because the probability of plant appearing increases with the introducing of balcony information.

(**a**) (**b**)

Figure 7. The detection results with scene-level information or without scene-level information. (**a**): without scene-level information. (**b**): with scene-level information.

Table 5. Ablation study on PASCAL VOC 2007 test. All methods are trained on PASCAL VOC 2007trainval+2012trainval. Baseline: faster region-based convolutional network (faster R-CNN). Scene: only using scene information. The bold characters represent the best result for each column.

Method	Map	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	Tv
Baseline	73.2	**76.5**	79.0	70.9	**65.5**	52.1	83.1	84.7	86.4	52.0	**81.9**	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	**83.0**	72.6
scene	**75.8**	75.9	**79.8**	**76.8**	61.8	**63.2**	**84.9**	**87.1**	**87.2**	**59.8**	81.3	**73.4**	**86.3**	**85.9**	**78.6**	**77.9**	**47.9**	**75.8**	**75.9**	82.3	**75.1**

Table 6. Ablation study on MS COCO 2014 minival. All methods are trained on MS COCO train set. Baseline: Faster R-CNN trained. Scene: only using scene information. The bold characters represent the best result for each column.

Method	AP	AP^{50}	AP^{70}	AP^{S}	AP^{M}	AP^{L}	AR^{1}	AR^{10}	AR^{100}	AR^{S}	AR^{M}	AR^{L}
Baseline	21.1	40.9	19.9	6.7	22.5	32.3	21.5	30.4	30.8	9.9	33.4	49.4
scene	**23.5**	**46.0**	**22.1**	**8.2**	**28.1**	**37.5**	**22.7**	**32.5**	**33.1**	**11.8**	**38.0**	**52.2**

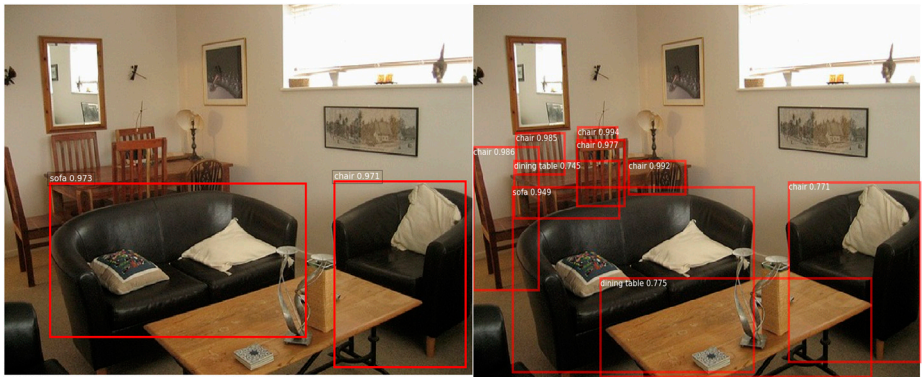
Effectiveness of Relation and Relation Settings: In the second ablation experiment, the validity of relationship information is evaluated. Here, the scene-level information is ignored in our model and we only use a set of Relation GRUs for object detection. Experiments are performed on the PASCAL VOC and MS COCO databases, respectively. From the experimental results shown in Tables 7 and 8, we can see that the performance of our model with only relationship information is still superior to the baseline (Faster R-CNN), especially for highly correlated objects. Taking the detection results in Figure 8 as an example. It is clear that due to the introduction of relation information, the tables and chairs which are strongly correlated with each other can be more accurately detected. This indicates the object relationship is very important for detection.

Table 7. Ablation study on PASCAL VOC 2007 test. All methods are trained on PASCAL VOC 2007trainval + 2012trainval. Baseline: Faster R-CNN. Relation: only using object-object relationships. The bold characters represent the best result for each column.

Method	Map	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	Tv
Baseline	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	**77.5**	76.7	38.8	73.6	**73.9**	**83.0**	72.6
Relation	**76.4**	**77.9**	**80.0**	**75.1**	**67.2**	**62.5**	**86.0**	**86.4**	**88.6**	**61.0**	**84.7**	**72.9**	**86.8**	**87.3**	77.4	**78.7**	**46.6**	**76.3**	72.9	82.2	**76.7**

Table 8. Ablation study on MS COCO 2014 minival. All methods are trained on MS COCO train set. Baseline: Faster R-CNN our trained. Relation: only using object-object relationships. The bold characters represent the best result for each column.

Method	AP	AP^{50}	AP^{70}	AP^{S}	AP^{M}	AP^{L}	AR^{1}	AR^{10}	AR^{100}	AR^{S}	AR^{M}	AR^{L}
Baseline	21.1	40.9	19.9	6.7	22.5	32.3	21.5	30.4	30.8	9.9	33.4	49.4
Relation	**24.5**	**47.4**	**23.1**	**8.7**	**29.7**	**38.5**	**23.1**	**33.3**	**33.9**	**12.6**	**39.5**	**52.5**



(**a**) (**b**)

Figure 8. The detection results with relation information or without relation information. (**a**): without relation information. (**b**): with relation information.

Through the above experiments, we can clearly know that both the scene-level information and object relationship are beneficial for detection, and they are indispensable. Figure 9 shows the detection results of some algorithms and our model. It can be seen that our model performs better when detecting small and highly correlated objects (such as driver, table, chair) due to the message

passing between scene-level information and object relationship. At the same time, for some objects with strong correlation with the scene, the detection result is also well (such as the boats in the sea scene and aeroplanes in the sky scene).

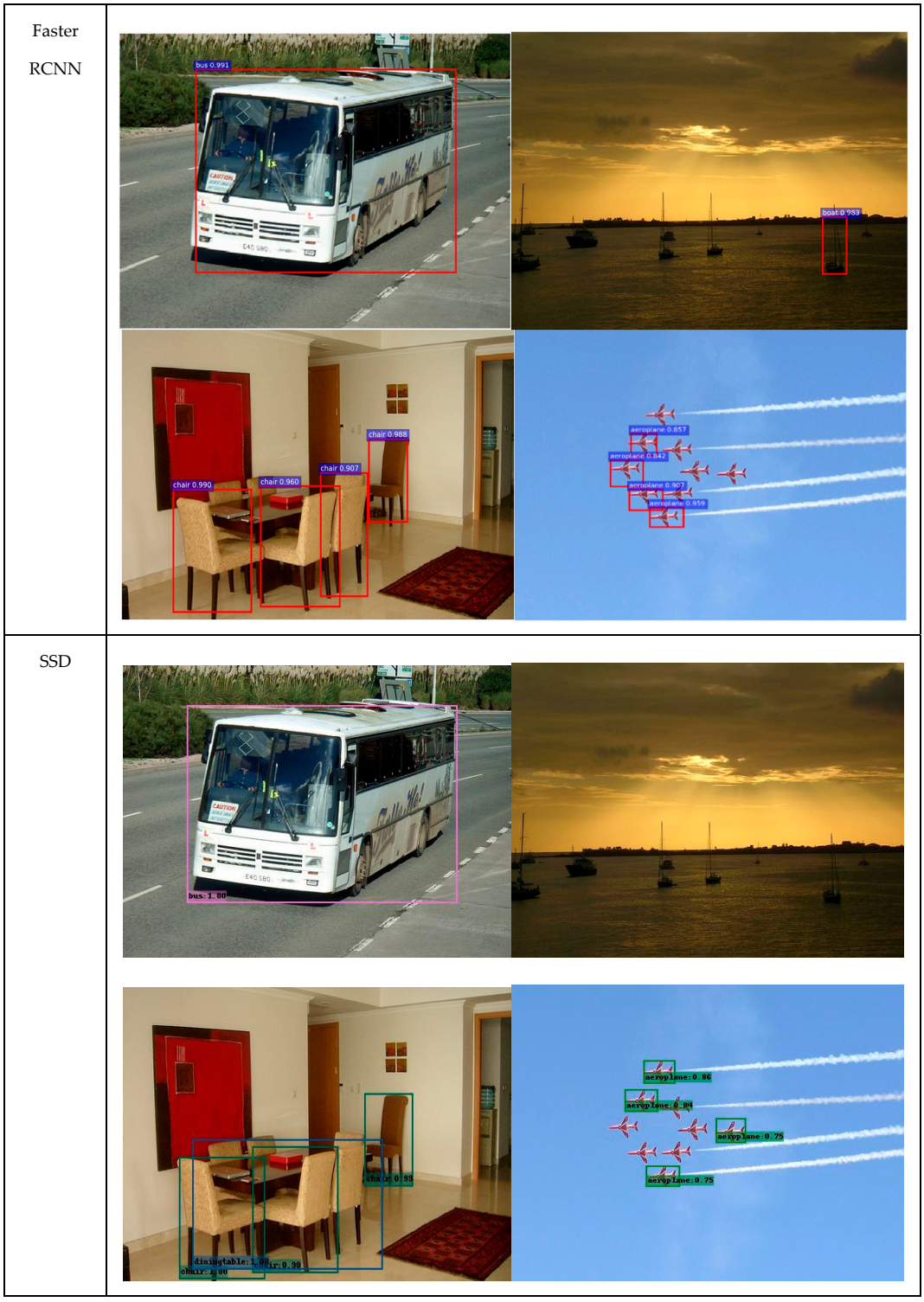


Figure 9. *Cont.*

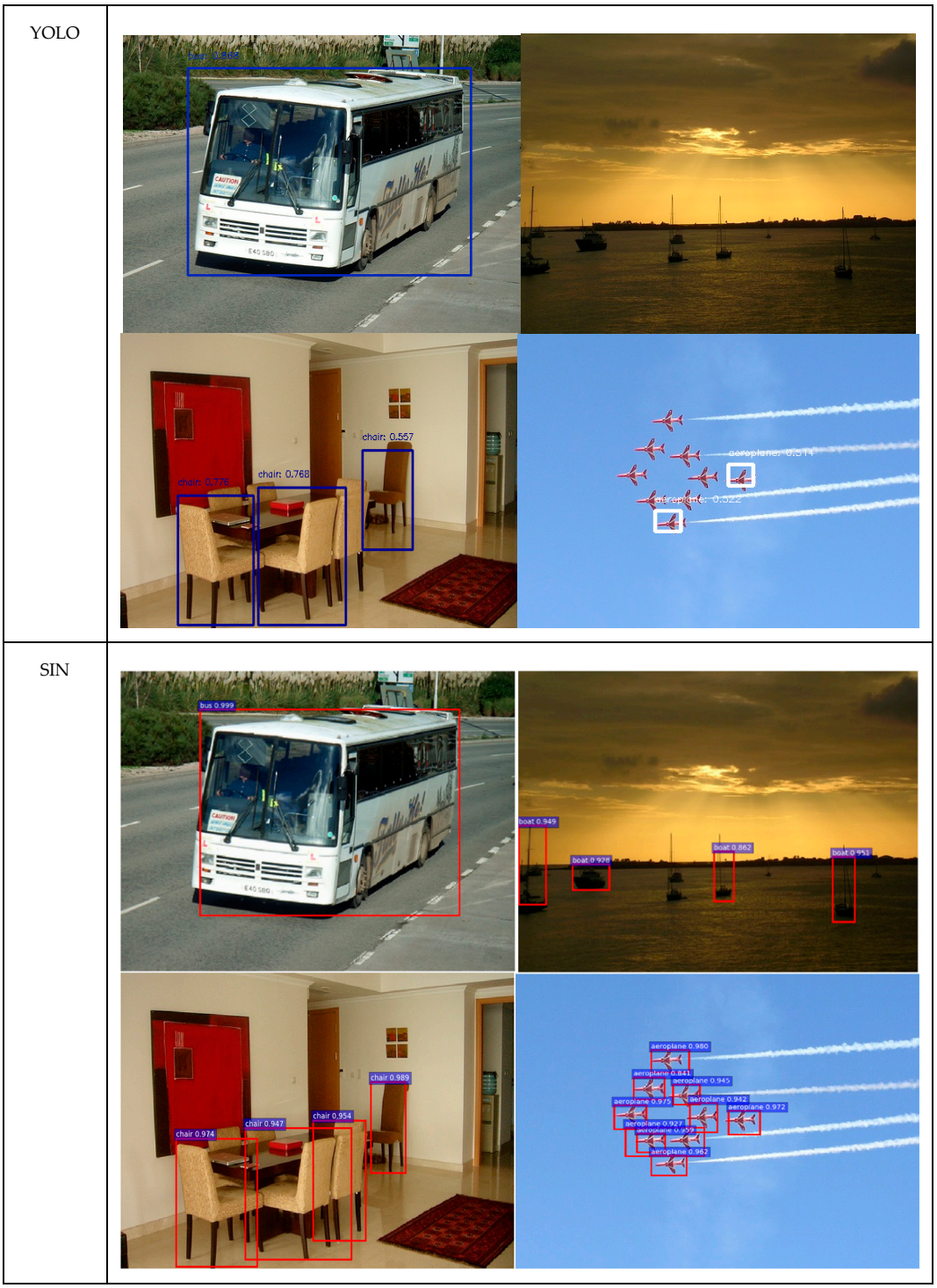


Figure 9. *Cont.*

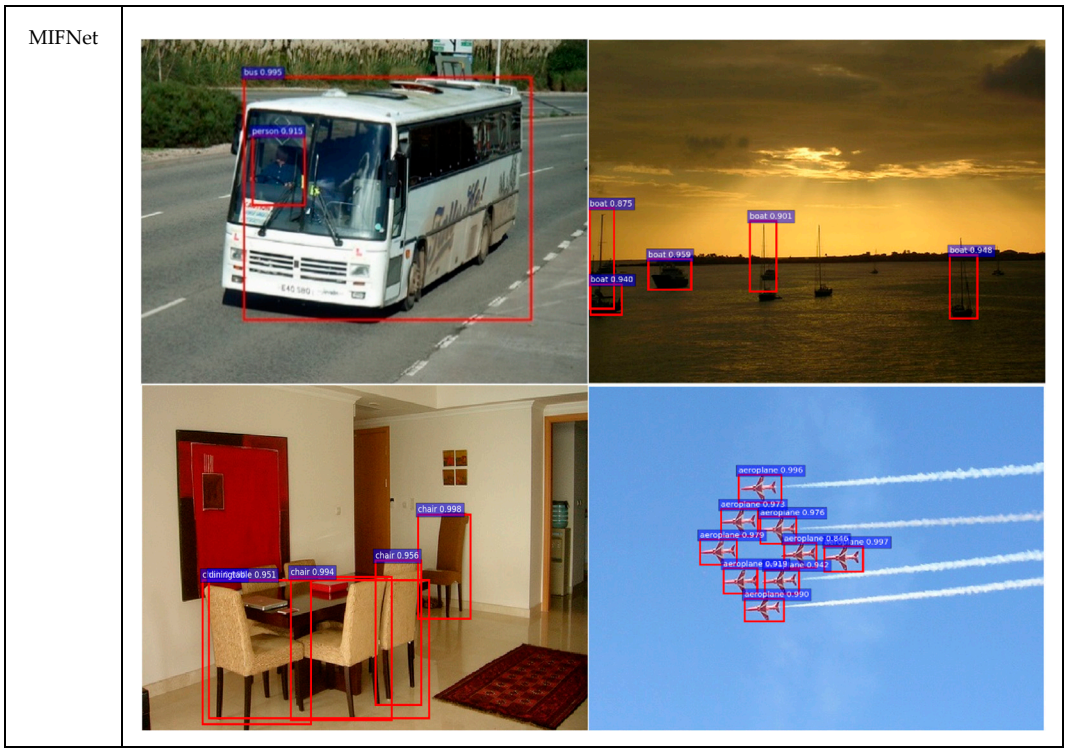


Figure 9. Comparison of detection results.

In order to test whether the number of relation modules will influence the detection result of our MIFNet, we also conduct an experiment to compare the performances of the proposed model with different numbers of relation modules. As shown in Table 9, we find that with the increase of the number of relation modules, the detection accuracy of our model will gradually decrease. This may because too many relation modules will make the network over-associate two objects. For example, once an object appears near the table, it will be detected as a chair regardless its feature. Therefore, we choose one module in the experiment.

Table 9. Comparison of different relation module settings. K × Relation denotes with K connected relation modules. The bold characters represent the best result for each column.

Database	GRU Settings	mAP
PASCAL VOC	1 × Relation	**77.6**
	2 × Relation	75.2
	3 × Relation	74.7
MS COCO	1 × Relation	**29.3**
	2 × Relation	24.0
	3 × Relation	24.0

Effectiveness of GRU Settings and the inputs of GRU: In our network, multiple parallel GRU units are used to fuse the information of scene-level context and object relationship. In order to study the effectiveness of different GRU settings, several experiments are conducted. Firstly, we build the message passing module with different numbers (1 to 3) of GRU units and test their performances. From the experimental results in Table 10, it can be found that when the number of stacked GRU units increases from 1 to 2, the mAP decreases. In addition, when the number of stacked GRU units increases from 2 to 3, no significant performance change can be observed. This indicates that one stacked GRU is enough for our proposed MIFNet.

Table 10. Comparison of different GRU settings utilized in the attention-based global context sub-module. The experiments are conducted on PASCAL VOC 2007. (K × GRU) denotes that there are K stacked GRU units. The bold characters represent the best result for each column.

GRU Settings	mAP	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	Tv
1 × GRU	**77.6**	**79.2**	79.8	**77.4**	**71.4**	63.3	86.3	87.1	**89.4**	**63.1**	**85.1**	**72.4**	86.7	**86.8**	78.2	79.0	**50.7**	**77.9**	75.4	**84.8**	**77.6**
2 × GRU	76.9	77.4	79.8	76.8	68.5	**63.7**	85.6	87.2	87.1	60.4	85.0	70.8	86.3	86.2	**79.3**	**79.3**	48.3	76.4	76.7	85.3	77.2
3 × GRU	76.9	76.5	**80.1**	77.2	67.9	62.8	**86.7**	86.5	87.2	**63.1**	84.0	71.9	87.0	86.1	77.8	78.7	47.8	76.8	**77.6**	84.1	76.1

Then, for the purpose of verifying the effectiveness of the message passing module in our MIFNet, we compare the experimental performances of two methods. The first method uses the scene-level information and the object relationship information as inputs to the different GRUs, which is the strategy employed in our approach. The second method is to concatenate the scene-level information and the object relationship information as one vector and then input this vector to only one GRU. From the experimental results in Table 11, the detection performances of two different methods are 77.6% and 76.2%, respectively. It is clear that the first method obtains better detection results since different information can be effectively transmit to each other through the two groups of GRUs in it. Nevertheless, the second method which directly concatenates different information cannot accomplish information transmission.

Table 11. Comparison of different inputs of GRU. The experiments are conducted on PASCAL VOC 2007. All methods are trained on PASCAL VOC 2007 trainval + 2012 trainval. GRU1 represents the inputs are the scene-level information and relationship with objects. GRU2 represents the mixed information. The bold characters represent the best result for each column.

Settings	mAP	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	Tv
GRU1	**77.6**	**79.2**	79.8	**77.4**	**71.4**	**63.3**	**86.3**	87.1	**89.4**	**63.1**	**85.1**	**72.4**	86.7	**86.8**	78.2	**79.0**	**50.7**	**77.9**	75.4	**84.8**	**77.6**
GRU2	76.2	76.4	80.7	74.9	65.9	62.1	83.7	**87.3**	87.8	61.6	82.4	67.7	**87.2**	84.7	**78.7**	78.4	49.9	77.6	**77.2**	81.6	77.1

5. Conclusions

This paper proposed a network that fuses both the scene-level and relationship information for object detection in images. Compared with other methods, the most important advantage of our approach is that we leverage the attention mechanism to model the relationship between objects adaptively. Besides, the relationship weights are obtained using not only the geometric structure, but also the appearance feature of the objects. At last, the scene-level information is also considered in our model. Two widely used databases are employed in our experiment. From the experimental results, we can see that through fusing the scene-level and object relationship information, our proposed MIFNet outperforms some classical and state-of-the-art approaches. Furthermore, some ablation experiments are also carried out to test the effectiveness of our MIFNet.

Author Contributions: Conceptualization, Y.Z. and J.W.; methodology, Y.Z. and J.W.; validation, Y.Z., J.W., and J.K.; formal analysis, M.Q.; data curation, Y.Z. and J.W.; software, Y.Z. and Y.L. (Yunpeng Liu); writing—original draft preparation, Y.Z.; writing—review and editing, J.W. and M.Q.; supervision, Y.L. (Yinghua Lu); project administration, J.K. and Y.L. (Yinghua Lu); funding acquisition, J.K. and Y.L. (Yinghua Lu). All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (Nos. 61672150, 61907007, 61702092), by the Fund of the Jilin Provincial Science and Technology Department Project (Nos. 20180201089GX, 20190201305JC) and the Fundamental Research Funds for the Central Universities (No. 2412019FZ049).

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

References

1. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1627–1645. [CrossRef] [PubMed]
2. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
3. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
4. Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. In *European Conference on Computer Vision*; Springer: Berlin, Germany, 2006; pp. 404–417.
5. Suykens, J.A.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [CrossRef]
6. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
7. Ren, S.; He, K.; Girshick, R.; Sun, J. *Faster r-cnn: Towards Real-Time Object Detection with Region Proposal Networks*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2015; pp. 91–99.
8. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
9. Zitnick, C.L.; Dollár, P. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 391–405.
10. Shi, W.; Bao, S.; Tan, D. FFESSD: An Accurate and Efficient Single-Shot Detector for Target Detection. *Appl. Sci.* **2019**, *9*, 4276. [CrossRef]
11. He, W.; Huang, Z.; Wei, Z.; Li, C.; Guo, B. TF-YOLO: An Improved Incremental Network for Real-Time Object Detection. *Appl. Sci.* **2019**, *9*, 3225. [CrossRef]
12. Pinheiro, P.O.; Collobert, R.; Dollár, P. Learning to segment object candidates. In Proceedings of the Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 1990–1998.
13. Jiang, Y.; Peng, T.; Tan, N. CP-SSD: Context Information Scene Perception Object Detection Based on SSD. *Appl. Sci.* **2019**, *9*, 2785. [CrossRef]
14. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
16. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
17. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In Proceedings of the Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 379–387.
18. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 354–370.
19. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
20. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
21. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
22. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv* **2016**, arXiv:1602.07261.
23. Huang, J.; Rathod, V. Supercharge your Computer Vision Models with the TensorFlow Object Detection API. Google AI Blog, 15 June 2017. Available online: https://ai.googleblog.com/2017/06/supercharge-your-computervision-models.html (accessed on 4 January 2020).

24. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the International Conference on Computer Vision & Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
25. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. *arXiv* **2017**, arXiv:1612.08242.
26. Najibi, M.; Rastegari, M.; Davis, L.S. G-cnn: An iterative grid based object detector. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2369–2377.
27. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
28. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
29. Jae Hwang, S.; Ravi, S.N.; Tao, Z.; Kim, H.J.; Collins, M.D.; Singh, V. Tensorize, factorize and regularize: Robust visual relationship learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1014–1023.
30. Krishna, R.; Chami, I.; Bernstein, M.; Fei-Fei, L. Referring relationships. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6867–6876.
31. Chen, X.; Li, L.J.; Fei-Fei, L.; Gupta, A. Iterative visual reasoning beyond convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7239–7248.
32. Bell, S.; Lawrence Zitnick, C.; Bala, K.; Girshick, R. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2874–2883.
33. 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.
34. Gkioxari, G.; Girshick, R.; Dollár, P.; He, K. Detecting and recognizing human-object interactions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8359–8367.
35. Liu, Y.; Wang, R.; Shan, S.; Chen, X. Structure inference net: Object detection using scene-level context and instance-level relationships. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6985–6994.
36. Hu, H.; Gu, J.; Zhang, Z.; Dai, J.; Wei, Y. Relation networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3588–3597.
37. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
38. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Conference and Workshop on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
39. Zeng, X.; Ouyang, W.; Yang, B.; Yan, J.; Wang, X. Gated bi-directional cnn for object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 354–369.
40. Shrivastava, A.; Gupta, A. Contextual priming and feedback for faster r-cnn. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 330–348.
41. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
42. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 740–755.
43. 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. [CrossRef]
44. Torralba, A. Contextual priming for object detection. *Int. J. Comput. Vis.* **2003**, *53*, 169–191. [CrossRef]
45. Oliva, A.; Torralba, A. The role of context in object recognition. *Trends Cognit. Sci.* **2007**, *11*, 520–527. [CrossRef]

46. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv* **2014**, arXiv:1412.7062.
47. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
48. Divvala, S.K.; Hoiem, D.; Hays, J.H.; Efros, A.A.; Hebert, M. An empirical study of context in object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, USA, 20–25 June 2009; pp. 1271–1278.
49. Galleguillos, C.; Belongie, S. Context based object categorization: A critical survey. *Comput. Vis. Image Underst.* **2010**, *114*, 712–722. [CrossRef]
50. Zhang, X.; Yang, Y.H.; Han, Z.; Wang, H.; Gao, C. Object class detection: A survey. *ACM Comput. Surv.* **2013**, *46*, 10. [CrossRef]
51. Ouyang, W.; Wang, X.; Zeng, X.; Qiu, S.; Luo, P.; Tian, Y.; Li, H.; Yang, S.; Wang, Z.; Loy, C.-C.; et al. Deepid-net: Deformable deep convolutional neural networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 Jun 2015; pp. 2403–2412.
52. Hu, D. An introductory survey on attention mechanisms in NLP problems. In *Proceedings of SAI Intelligent Systems Conference*; Springer: Cham, Switzerland, 2019; pp. 432–448.
53. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the International Conference on Machine Learning, Ithaca, NY, USA, June 2015; pp. 2048–2057.
54. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
55. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
56. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv* **2016**, arXiv:1603.04467.
57. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
58. 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 Beach, FL, USA, 20–25 June 2009; pp. 248–255.