applied sciences

MDPI

Article

Object Detection Algorithm Based on Multiheaded Attention

Jie Jiang [1], Hui Xu [1,*], Shichang Zhang [2] and Yujie Fang [1]

[1] College of Systems Engineering, National University of Defense Technology, Changsha 410073, China; JieJiang@nudt.edu.cn (J.J.); 17719498319@163.com (Y.F.)
[2] School of Information Science and Engineering, Ocean University of China, Songling Road No. 238, Qingdao 266100, China; 18354226160@163.com
* Correspondence: xuhui17@nudt.edu.cn

Received: 14 April 2019; Accepted: 29 April 2019; Published: 2 May 2019

Abstract: This study proposes a multiheaded object detection algorithm referred to as MANet. The main purpose of the study is to integrate feature layers of different scales based on the attention mechanism and to enhance contextual connections. To achieve this, we first replaced the feed-forward base network of the single-shot detector with the ResNet–101 (inspired by the Deconvolutional Single-Shot Detector) and then applied linear interpolation and the attention mechanism. The information of the feature layers at different scales was fused to improve the accuracy of target detection. The primary contributions of this study are the propositions of (a) a fusion attention mechanism, and (b) a multiheaded attention fusion method. Our final MANet detector model effectively unifies the feature information among the feature layers at different scales, thus enabling it to detect objects with different sizes and with higher precision. We used the 512×512 input MANet (the backbone is ResNet–101) to obtain a mean accuracy of 82.7% based on the PASCAL visual object class 2007 test. These results demonstrated that our proposed method yielded better accuracy than those provided by the conventional Single-shot detector (SSD) and other advanced detectors.

Keywords: object detection; attention mechanism; multiheaded attention

1. Introduction

Target detection is a fundamental, challenging, and long-standing problem, and has been a hotspot in the field of computer vision research for decades [1–3]. The purpose of target detection is to determine if any instances of a specified category exist in a given image. If there is an object to be detected in a specific image, target detection returns the spatial position and the spatial extent of the instances of the objects (based on the use a bounding box, for example). As one of the cornerstones of image understanding and computer vision, target detection forms the basis for more complex or higher-level visual tasks, such as object tracking, image capture, instance segmentation, and others. In addition, target detection is also extensively used in areas such as artificial intelligence and information technology, including machine vision, automatic driving, and human–computer interaction.

Recently, the method of automatic learning of represented features from data based on deep learning has effectively improved the performance of target detection. Neural networks are the foundation of deep learning. Therefore, the design of better neural networks has become a key issue toward the improvement of target detection performance. Recently developed target detectors that have been based on convolutional neural networks (CNNs) have been classified in two types: The first is the two-stage detector type, such as Region-Based CNNs (R–CNNs) [4], Region-Based Full Convolutional Networks (R–FCNs) [5], and Feature Pyramid Networks (FPNs) [6], and the other is the single-stage detector, such as the You Only Look Once (YOLO) [1], Single-shot detector (SSD) [2],

and the RetinaNet [7]. The former type generates a series of candidate frames as samples, and then classifies the samples based on a CNN; the latter type does not generate candidate frames but directly converts the target frame positioning problem into a regression processing problem.

To maintain real-time speeds without sacrificing precision in the various target detectors described above, Liu et al. [2] proposed the SSD which is faster than YOLO and has a comparable accuracy to that of the most advanced region-based target detectors. SSD combines the regression idea of YOLO with the anchor box mechanism of a Faster R–CNN [3], predicts the object region on the feature maps of the different convolution layers, and outputs discretized multiscale and multi proportional default box coordinates. The convolution kernel predicts the frame coordinate compensation of a series of candidate frames and the confidence of each category. The local feature maps of the multiscale area are used to obtain results for each position of the entire image. This maintains the fast characteristics of the YOLO algorithm and also ensures that the frame positioning effect is similar to that induced by the Faster R–CNN [3]. However, SSD directly and independently uses two layers of the backbone (VGG16) and four extra layers obtained by a convolution with stride 2 to construct the feature pyramid but lacks strong contextual connections.

To solve these problems, we propose in this study a new, single-stage detection architecture, commonly referred to as MANet, which aggregates feature information at different scales. Our MANet achievs 82.7% mAP on the PASCAL VOC 2007 test [8].

The contributions of our work can be summarized as follows:

(1) We propose a framework referred to as MANet that combines feature information at different scales for better performance
(2) To fuse feature information at different scales, we propose a new attention mechanism referred to as the fusion attention
(3) To fuse multiple scale feature information in a more efficient manner, we designed different multihead fusion modules to generate more efficient feature representations

2. Related Studies

Prior to the advent of CNNs, early target detection methods were usually based on sliding windows, such as Support Vector Machines with a Histogram of Oriented Gradients (HOG–SVM) [9], or Deformable Part Models (DPMs) [10]. Most of them have been extensively used to classify Regions-Of-Interest (ROIs) into various categories. DPM [10] is one of the most commonly used methods. The model was proo-stage detecposed by Felzenszwalb et al. in 2008 [10]. As its name suggests, it is a component-based detection algorithm. It won the PASCAL VOC target detection championship three times. The algorithm is based on the extraction of the artificial features of DPM followed by the use of latent SVM classification. This feature extraction method has obvious limitations. Firstly, the DPM feature is computationally complex and the calculation speed is low. Secondly, the artificial feature yields poor detection outcomes on objects with rotation, stretching, and viewing angle changes. These drawbacks largely limit the application scenarios of the algorithm. To solve this problem, Girshick et al. proposed a target detection method based on a deep R–CNN [4] that exceeded the performances of the previous methods. Ever since the introduction of this method, almost all of the optimal target detection methods have been based on the CNN. Two-stage detectors and single-stage detectors are currently the two mainstream target detection methods. Two-stage detector examples include the Spatial Pyramid Pooling (SPPNet) [11], Fast R–CNN [12], Faster R–CNN [3], and the R–FCN [5]. Most of them use the top layer of the convolutional network to detect objects of different sizes. These methods first use a separate generator to generate a set of candidate objects, such as Selective Searches [13], Edge Boxes [14], and Region Proposal Network (RPN). Most of them are either based on super-pixel merging (such as the Constrained Parametric Min-Cuts (CPMC) [15]), or on sliding window algorithms (such as the Edge Boxes [14]). Their common feature is the CNN which is mainly used as a classifier. Object boundaries are not predicted. While these methods greatly improve the detection accuracy, they are computationally intensive and slow.

Single-stage detectors (such as YOLO [1] and SSD [2]) have drawn attention because of the shortcomings of the slower two-stage detectors. These detectors replace the region proposal stage by covering the entire image with a fixed set of anchors of different sizes. Small objects are detected in shallow convolution layers with high-resolution features, while large objects are detected in deep convolution layers with low-resolution features. Therefore, the SSD can extract rich features with smaller input sizes to reduce the computational cost compared to the Faster R–CNN [3]. However, this method type does not pay attention to the local information of each location, and the information in the lower convolution layer is not fully utilized. Therefore, although the single-stage method is less expensive in reference to the calculation compared to the two-stage method, the accuracy is still inadequate.

To improve the performance of two- and single-stage detectors, the academic community has proposed various strategies to aggregate multiscale information into target detectors. Multiscale CNN (MS–CNN) [16] utilizes feature maps at various resolutions to detect objects at different scales, thus resulting in a set of variable receptive field sizes which are used to cover many object sizes. Rainbow–SSD (R–SSD) [17] combines the characteristics of different layers by pooling and deconvolution that not only strengthens the relationship between different layer feature maps but also increases the number of these different layer feature maps. The Feature Fusion Single Shot Multibox Detector (FSSD) [18] takes each feature concat level and generates a feature pyramid from it. Enhanced SSD (ESSD) [19] fuses feature maps of different output layers and proposes a visual reasoning method to enhance the network.

In consideration of the small-object detection problem, the Deconvolutional Single-Shot Detector (DSSD) [20] uses an additional deconvolution layer to increase the resolution of the feature mapping layer and to fuse the context information. StairNet [21] introduced a combined feature module that enhances contextual semantic information in a top-down manner, further inferring the combined information. Feature-Fused SSD (Fast Detection for Small Objects) [22] improve accuracy by fusing feature maps with adjacent layers.

In contrast to previous studies, we propose herein a new architecture to achieve the fusion of global context information of different layers, thus improving the performance of target detection. Compared with the former fusion method, our strategy is more effective. It directly uses a new attention mechanism to fuse the features of different layers. At the same time, the multihead structure allows the model to include the character information from different scales that overcomes the feature-lacking shortages of the previous methods [18–20].

3. Methods

In this section, we first explain the concept of a multihead fusion feature. We then introduce the detailed structure of MANet. In Section 3.2.3, the construction of fusion attention is introduced. In Section 3.2.4, we introduce and deduce the structure and formulas of three multihead models

3.1. Multihead Fusion Feature

Recently, some network models have improved the accuracy of detection by combining feature maps with other layers (FSSD [18], ESSD [19], and others). This shows that feature fusion is beneficial to the improvement of the target detection performance. Therefore, we propose the concept of multihead fusion features and integrate the feature maps at different sizes by using multihead attention to obtain fusion features to extract more accurate information.

Luo et al. [23] indicated that the effective receptive field size is much smaller compared to the theoretical receptive field size. Therefore, the SSD model performs poorly on object detection applications. We suspect that the reason for which SSD cannot properly detect objects is the low-field-size of the feature layer. The receptive field is far smaller than the theoretical receptive field size. Therefore, it is possible to enhance the effective receptive field and strengthen the contextual connection by merging the feature map information from different depths and different sizes to improve the detection

performance. Therefore, we propose a long-term attention type, which is used to fuse the feature map information at different resolutions from different layers to make up the information of the current layer, thereby improving the performance of detection.

3.2. MANet Model

We first review the SSD structure. To improve the training efficiency, the basic MANet network adopts the same strategy as that adopted by DSSD [20] and replaces the Visual Geometry Group (VGG) [24] network used in the original SSD with the residual network. We then discuss the construction of a multiheaded network by adding fusion attention to obtain the semantic context information of the final MANet model.

3.2.1. VGG

The SSD [2] is a target detection algorithm proposed by Liu et al. at the 2016 European Computer Vision Conference (ECCV2016). It uses the same direct return methods of the bounding box (bbox) and classification probability in YOLO. It also applies the same technique used in the Faster R–CNN [3], and uses anchors extensively to improve the recognition accuracy. By combining these two structures, the SSD maintains a high-recognition speed and increases its average accuracy level. At the same time, the feature pyramid structure is used to predict the detection results, that is, conv4_3, conv_7 (FC7), conv6_2, conv7_2, conv8_2, and conv9_2, are used to detect feature maps with different sizes. Softmax classification and position regression are performed simultaneously on multiple feature maps. At the same time, the SSD does not use maximum suppression (NMS) to process the predicted results and remove redundant detection frames to obtain the final test results. More details on this process can be found in Reference [2], which uses the VGG as the underlying network.

3.2.2. ResNet

To train the network model in a more effective manner, we herein adopt the same strategy as that used for DSSD [20] (the performance of the residual network is better than that of the VGG network). The goal is to improve accuracy. However, the first implemented modification was the replacement of the VGG network used in the original SSD with ResNet [25]. The feature map has the same size (see Table 1 for details). Figure 1 shows the ResNet–SSD which uses the ResNet–101 [25] as the underlying network. As shown in Figure 1, we select conv3_x, conv5_x, conv6_x, conv7_x, conv8_x, and conv9_x, in the prediction module in ResNet–SSD, which is the similar manner as SSD. We also added a series of convolution feature layers at the end of the underlying network. These feature layers were gradually reduced in size that allowed prediction of the detection results on multiple scales. When the input size is 300 and 320, although the ResNet–101 layer is deeper than the VGG–16 layer, it is experimentally known that it replaces the SSD's underlying convolution network with the residual network, and it does not improve its accuracy but rather decreases it. We then gradually added different modules to improve the accuracy of the test.

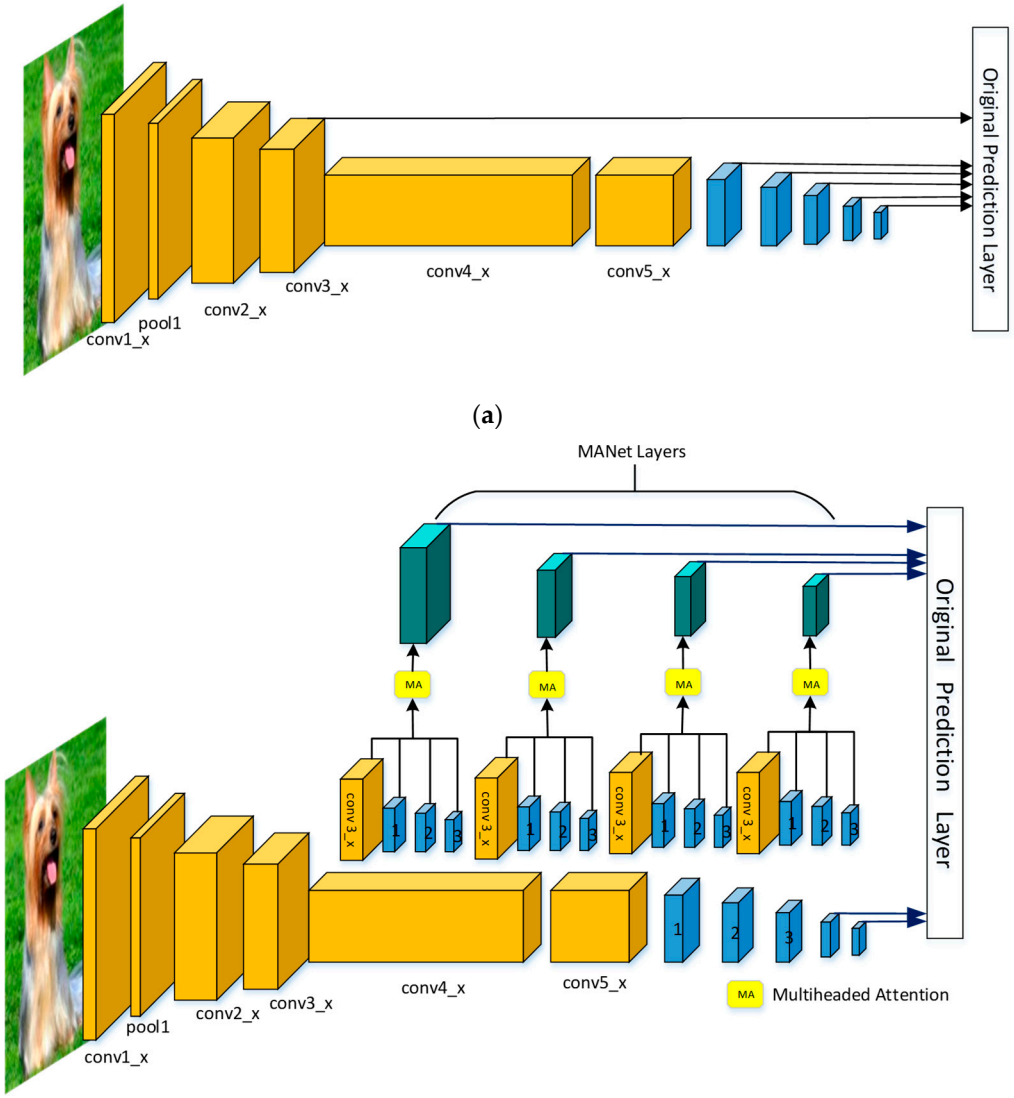

Figure 1. Networks of (**a**) Single-shot Multibox Detector (SSD) and (**b**) Multiheaded Attention Net (MANet). The blue modules are the layers added in the SSD framework are referred to as the SSD layers. In (**b**), the green layers are the MANet layers (where conv1_x, conv2_x, conv3_x, conv4_x, conv5_x, are the convolution feature layers at different scales).

3.2.3. Fusion Attention

In the execution of the task of target detection, it can be intuitively understood that in the CNN, the shallow feature maps tend to be small. Therefore, they are usually used for the detection of small objects, and the feature layers that are closer to the top layer are often very sensitive. Therefore, large feature maps are often used for the detection of large objects. In recent years, many jobs have enhanced the range of receptive fields by combining the feature information from feature maps obtained at different scales. For example, in 2017, Fu et al. proposed the DSSD [20], the first constructed SSD based on ResNet–101. The authors introduced a deconvolution layer as the code–decoder to convey contextual information, thus achieving excellent results in small-object detection. In essence, the feature information of different receptive fields was fused by deconvolution. Therefore, the relationship among the feature maps of receptive fields with different sizes has an important influence on the target detection accuracy. The relationship among the feature maps of receptive fields with different sizes is intended to capture the dependencies among the different feature maps associated with the

different receptive fields and enhance their respective feature expression capabilities. Therefore, to obtain the feature information from the various receptive field maps with different sizes, this study introduces a fusion attention module that mainly uses the correlation between different scale feature maps to enhance the expression of each feature. We describe below the working process of the fusion attention module.

As shown in Figure 2, the feature map $X \in R^{C\times H\times W}$ is first input to the convolution layer with the batch normalization and the ReLU layers, generates two new features F and G—where $C \times H \times W$ represent the channel, height, width, $\{F, G\} \in R^{C\times H\times W}$—and then reshapes them to $R^{C\times N}$, where $N = H \times W$ denotes the number of features. The features F and G are then transposed and multiplied, and are then normalized by the softmax operation to obtain the attention map of the scale feature $O \in R^{N\times N}$:

$$f_{ji} = \frac{exp\left(F_i \cdot G_j\right)}{\sum_{i=1}^{N} exp\left(F_i \cdot G_j\right)} \tag{1}$$

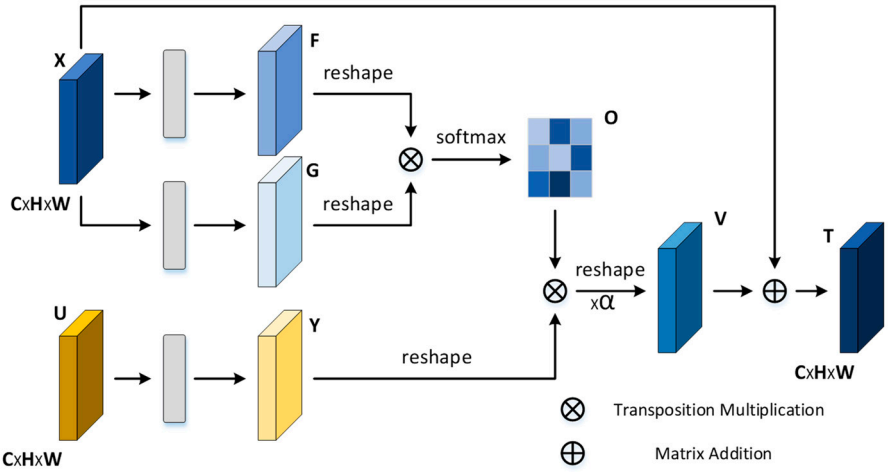

Figure 2. Structure of the fusion attention module.

At the same time, another feature map $W \in R^{C\times H\times W}$ is constructed, and the feature U is sent to the convolution layer with the batch normalization and ReLU layers to generate a new feature $Y \in R^{C\times H\times W}$, shape it into $R^{C\times N}$, multiply the attention map O by the transpose of the feature Y, and reshape the result to $R^{C\times H\times W}$. Finally, the output is multiplied by a scale parameter α to obtain V_j and a sum operation is executed with the feature X to obtain the final output $T \in R^{C\times H\times W}$ as follows,

$$\begin{aligned} V_j &= \alpha \sum_{i=1}^{N} \left(f_{ji} Y_i\right) \\ T_j &= \alpha \sum_{i=1}^{N} \left(f_{ji} Y_i\right) + X_J \end{aligned} \tag{2}$$

where α is initialized to zero and is gradually assigned to additional weights. It can be inferred from the above formula that the final feature T is the result of the fusion of the two different feature maps.

Thus, the fusion attention module has the ability to fuse different features and selectively aggregate features. A mutual gain between the features is achieved and is more conducive to target detection.

3.2.4. Multiple Modules

Feature maps of different resolutions contain different feature information. SSD [2] directly uses the features of the two-layer skeletal network (i.e., VGG16) and the four additional layers obtained by the convolution with stride 2 to construct the feature pyramid. So we suspect that each feature map in the feature gold tower contains insufficient feature information to cause poor detection performance.

Therefore, we can make up the information of the current layer by merging the information from other scale feature maps. In this study, we constructed a multiheaded attention module that was used to (a) fuse information among different scales, and (b) enhance the feature expression ability of different global scales to improve the accuracy of target detection.

To reduce the number of calculations, we have eliminated two small scales (1×1, 3×3) from the multiheaded module, and we directly input these in the final prediction module. As shown in Figure 3, we used the 5×5 feature map as an example, and we designed three different multihead fusion models to fuse the features of the various scales. To facilitate the calculation, we integrated the features of the different scales. First, the sizes of the feature maps of the different scales were unified to the same scale. To reduce the computation, the unified scale operation was directly executed using bilinear interpolation. No parameters were introduced, the number of calculations was small, and the channel was then unified to the same dimension based on 1×1 convolution. The details of the three multihead models are as follows:

(1) Model a performs the concatenation operation on the integrated feature tensor, and then directly inputs it into the fusion attention module to obtain the final fusion result. Equation (3) was used, where f_{all} denotes the new feature outcome after the application of the concat operation on the three features, and *FA* is the fusion attention module.

$$\begin{aligned} f_{\text{all}} &= concat(f_{38\times38}, f_{19\times19}, f_{10\times10}) \\ output_{\text{a}} &= FA(f_{5\times5}, f_{\text{all}}) \end{aligned} \tag{3}$$

(2) Model b inputs the integrated feature tensors into the fusion attention module, and then concatenates the output results to obtain the final outcome according to Equation (4), where f_1, f_2, f_3, indicate that the features of different scales are merged with the current feature scales by the fusion attention module to obtain new features, and f_1, f_2, f_3, are concatenated to obtain the final result.

$$\begin{aligned} f_1 &= FA(f_{5\times5}, f_{38\times38}) \\ f_2 &= FA(f_{5\times5}, f_{19\times19}) \\ f_3 &= FA(f_{5\times5}, f_{10\times10}) \\ output_{\text{b}} &= concat(f_1, f_2, f_3) \end{aligned} \tag{4}$$

(3) Model c is different from the previous two models. Instead of directly using the fusion attention module, V is correspondingly calculated at different scales, and the results are concatenated and then compared with the feature scale of the layer, which is added to yield the final $output_{\text{c}}$ Equation (5) is used as follows, V_j

$$\begin{aligned} V_1 &= V(f_{5\times5}, f_{38\times38}) \\ V_2 &= V(f_{5\times5}, f_{19\times19}) \\ V_3 &= V(f_{5\times5}, f_{10\times10}) \\ V_{\text{c}} &= concat(V_1, V_2, V_3) output_{\text{c}} = V_{\text{c}} + f_{5\times5} \end{aligned} \tag{5}$$

where V is the V_j in the fusion attention module.

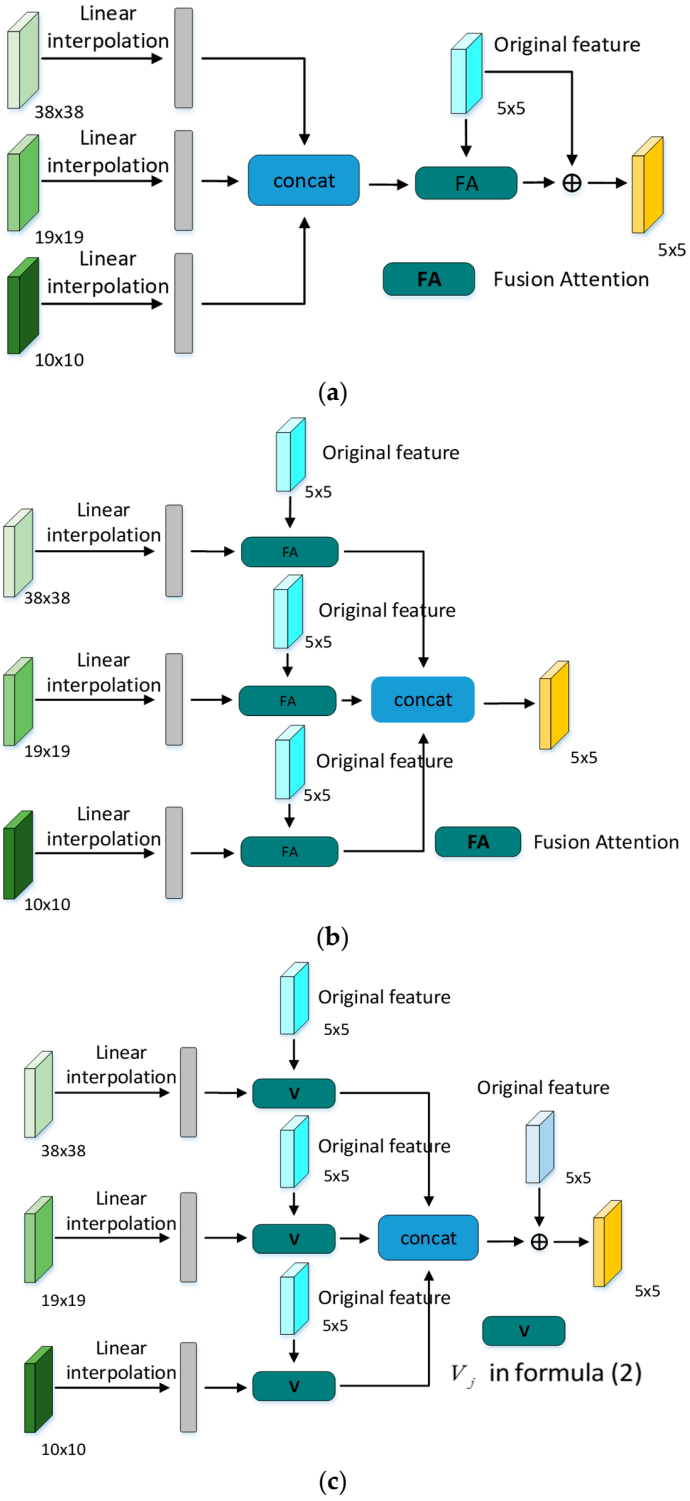

Figure 3. Variants of the multihead module. (**a**) Model a; (**b**) Model b; (**c**) Model c; The three multihead fusion models that fuse various scale features. Model c is our final choice.

4. Experiments

We evaluated our method on the PASCAL VOC [8] dataset, which had 20 object categories. According to the PASCAL VOC protocol [8], we conducted joint training on the VOC 2007 and the VOC 2012 training validation sets and performed tests on the VOC 2007 test set. We evaluated the model using standard mean accuracy (mAP) scores. To improve the efficiency of the experiment, we

used ResNet–50 [25] as the feed-forward convolutional neural network in the ablation experiment. In addition to the techniques included in the SSD, we did not use other techniques.

4.1. Base Network

Our final experimental used the ResNet–101 [25] as the feature extraction network, which was pretrained on the Imagenet [26] dataset. We changed the effective stride of the conv_5 stage from 32 pixels to 16 pixels to increase the feature map resolution. Following the application of the SSD and the fitting of the residual architecture, we used residual blocks to add a few extra layers by decreasing the feature map size.

Table 1 shows the feature layers selected in the original VGG architecture and ResNet–101. Depth refers to the location of the selected feature layer in the network (only the convolutional and the pooled layers were considered). It is important to note the depths of the first prediction layers in the two networks. While ResNet–101 contained 101 layers, we need to use a dense feature layer to predict smaller targets. Therefore, we can only choose the last feature layer in conv3_x as the first prediction layer. If we only consider kernel sizes greater than one, values up to nine can be considered. This means that the receptive field of neurons in this layer may be smaller than the receptive field of the conv4_3 neurons in VGG. Compared to the other layers of ResNet–101 [25], the layer has a weak feature expression and a poor prediction performance.

Table 1. Selected feature layers in VGG and ResNet–101.

VGG	**Conv4_3**	**Conv7**	**Conv8_2**	**Conv9_2**	**Conv10_2**	**Conv11_2**
Resolution	38 ×38	19 ×19	10 ×10	5 ×5	3 ×3	1
Depth	13	20	22	24	26	27
ResNet–101	**Conv3_x**	**Conv5_x**	**Conv6_x**	**Conv7_x**	**Conv8_x**	**Conv9_x**
Resolution	40 ×40	20 ×20	10 ×10	5 ×5	3 ×3	1
Depth	23	101	104	107	110	113

4.2. Results on Pascal VOC 2007

MANet was developed with PyTorch v0.4.1. We conducted experiments on a machine with 2 NVIDIA 1080Ti graphics processing units (GPUs), CUDA8.0 and cuDNN 6.0.21. The input sizes were set to 300 and 320. Owing to memory limitations, we set the batch size to 32 to train the model. When the input size is 512, the batch size is set to 12 to train the model. Similar to the original SSD [2] model, we used a weight attenuation of 0.0005 and a momentum of 0.9. First, a stochastic gradient descent (SGD) [27] optimizer with an initial learning rate of 0.001) was used, but the training process was not so stable owing to the abrupt loss fluctuations. To solve this problem, we used a warm-up strategy by training the model at a learning rate of 0.001 in the first two epochs (an epoch refers to the process of completing a forward calculation and backpropagation in all data feeds into the network). After the warm-up phase, the process returned to the original learning rate plan, which was reduced by a factor of 10 at the 150th and 200th epochs. Training was executed up to the 250th epoch before the process was terminated.

Table 2 shows the results of our tests on the PASCAL VOC 2007 test set. We set the batch size to 1, take the sum of the CNN time and NMS time of 4000 images, and divide by 4000 to get the inference time of a single image. For fair comparison, we reproduce and test the speed of SSD300-VGG16 on our device. By replacing the VGG [24] with ResNet–101 [25], the performance becomes lower than the original SSD (input resolution size is 320 and 300). This may be related to the capability of the feature layer. The original SSD only uses the feature maps at the different depths for prediction and does not fuse the low-level and high-level features. We used the original SSD as the baseline of our experiment. We know from the table results that we have improved the detection rate on the basis of the original SSD512 [2] (the backbone network is VGG16) by 2.9%. At the same time, our MANet320 can achieve

a mAP of 80.2%, which is improved by 1.6% compared with the DSSD321. In addition, when our image input size is 512, MANet leads to some improvement compared with other multiscale fusion methods (such as FSSD [18] and ESSD [19]). This proves the effectiveness of MANet. To compare the performance of SSD and MANet in an intuitive manner, as shown in Figure 4, we randomly sampled a certain number of images from the PASCAL VOC test set and compared the outcomes.

Table 2. Comparisons of speed and accuracy based on PASCAL VOC2007 tests (We have improved on the basis of SSD, only improved the network structure, and did not introduce other data enhancement methods other than the data enhancement methods mentioned in SSD).

Method	Backbone	Input Resolution	GPU	FPS	mAP (%)
Two-stage					
Fast R–CNN [12]	VGG16	1000 × 600	Titan X	0.5	70.0
Faster R–CNN [3]	VGG16	1000 × 600	Titan X	7.0	73.2
OHEM [28]	VGG16	1000 × 600	Titan X	7.0	74.6
HyperNet [29]	VGG16	1000 × 600	Titan X	0.9	76.3
ION [30]	VGG16	1000 × 600	Titan X	1.3	76.5
Faster R–CNN [3]	ResNet–101	1000 × 600	K40	2.4	76.4
R–FCN [5]	ResNet–101	1000 × 600	Titan X	9.0	80.5
CoupleNet [31]	ResNet–101	1000 × 600	Titan X	8.2	82.7
Single-stage					
YOLO [1]	GoogleNet	448 × 448	Titan X	45.0	63.4
YOLOv2 [32]	DarkNet19 [32]	544 × 544	Titan X	40.0	78.6
DSOD300 [33]	DenseNet [34]	300 × 300	Titan X	17.4	77.7
SSD300 [2]	VGG16	300 × 300	Titan X	46.0	77.2
SSD300 [2]	VGG16	300 × 300	1080Ti	71.0	77.7
SSD512 [2]	VGG16	512 × 512	Titan X	19.0	79.8
SSD321 [20]	ResNet–101	321 × 321	Titan X	11.2	77.1
SSD513 [20]	ResNet–101	513 × 513	Titan X	6.8	80.6
DSSD321 [20]	ResNet–101	321 × 321	Titan X	9.5	78.6
DSSD513 [20]	ResNet–101	513 × 513	Titan X	5.5	81.5
Feature-fused SSD [22]	VGG16	300 × 300	Titan X	43.0	78.9
R–SSD300 [17]	VGG16	300 × 300	Titan X	35.0	78.5
R–SSD512 [17]	VGG16	512 × 512	Titan X	16.6	80.8
FSSD300 [18]	VGG16	300 × 300	1080Ti	65.8	78.8
FSSD512 [18]	VGG16	512 × 512	1080Ti	35.7	80.9
ESSD300 [19]	VGG16	300 × 300	-	52.0	79.2
ESSD512 [19]	VGG16	512 × 512	-	18.6	82.4
MANet300	VGG16	300 × 300	1080Ti	50.0	79.1
MANet300	ResNet-101	300 × 300	1080Ti	37.0	79.4
MANet320	ResNet–101	320 × 320	1080Ti	34.0	80.2
MANet512	ResNet–101	512 × 512	1080Ti	25.0	82.7

4.2.1. Multiheaded Attention

To further verify the effectiveness of our method, we inserted the multi-attention module directly into the original SSD. From Table 2, we know that when our backbone network is VGG16, we have achieved good results. Our MANet300 achieved a 79.1% mAP, which effectively represents an increase of 1.9% compared to the original SSD, and increases of 0.6% and 0.3% compared to other fusion methods (RSSD [17] and FSSD [18]), thus further demonstrating that our fusion strategy is effective.

4.2.2. Small Objects Detection

We manually screened 541 images which contained small objects from the PASCAL VOC 2007 test set to better evaluate our model. Our MANet300 (to ensure the fairness of the experiment, the backbone network selected VGG16), improved its mAP response by 2.6% compared to the original SSD300. In addition, we visualized some of the results, as shown in Figure 5. From the figure, we can clearly see that our method is beneficial to the detection of small targets.

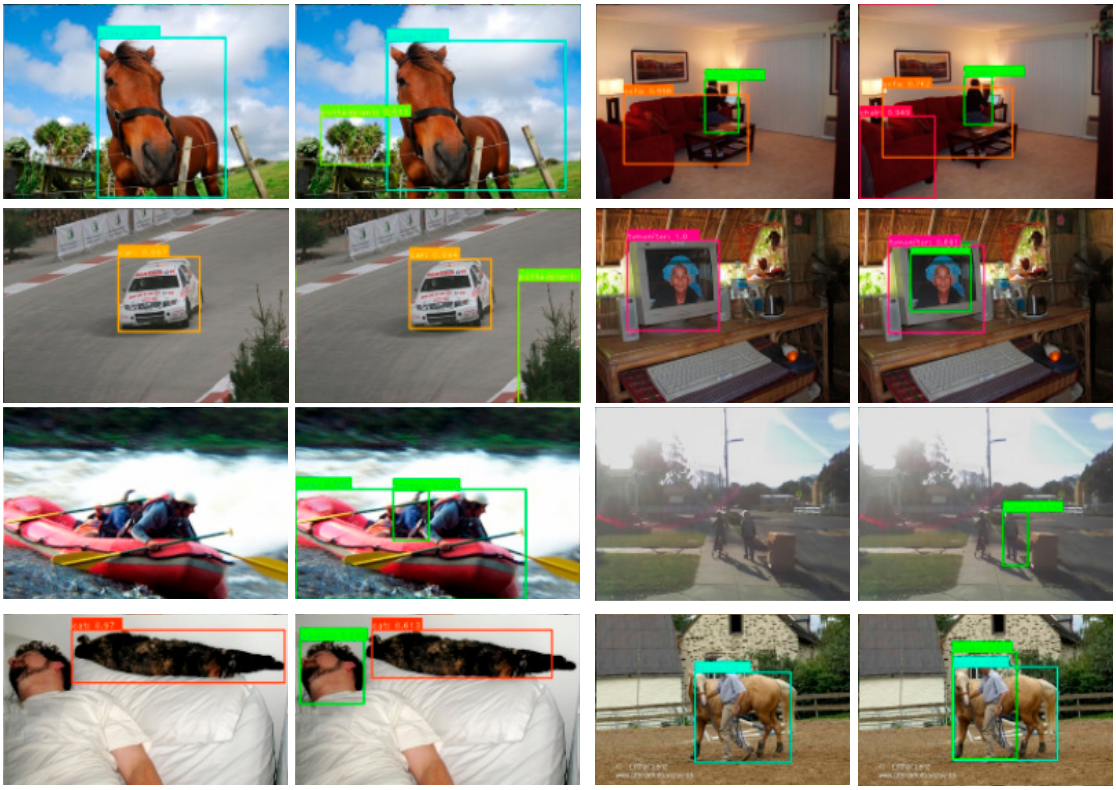

Figure 4. Detection examples on the VOC 2007 test-dev with the SSD320/MANet 320 model. For each pair, the images on the left are the results of the SSD, and the images on the right are the results of MANet.

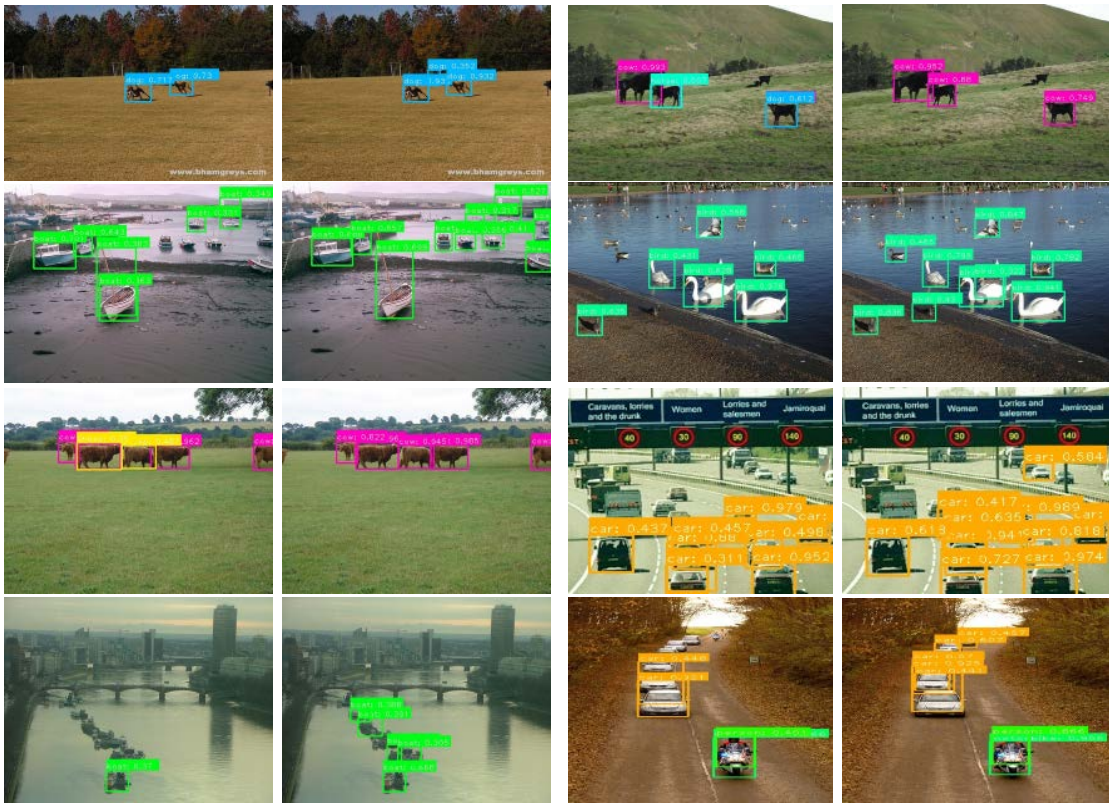

Figure 5. Detection examples on the VOC 2007 test-dev with the SSD300/MANet 300 model. For each pair, the images on the left are the results of the SSD, and the images on the right are the results of MANet.

4.3. VOC 2007 Ablation Study

In this section, to verify the impacts of each of the module on performance responses, we set up different models on the VOC 2007 dataset for testing. To improve the efficiency of the experiment, we used ResNet–50 [25] as the basic network of the model and tested it. The results are reported in Tables 3 and 4.

4.3.1. Variants of Multiple Models

To integrate the feature maps of different receptive fields in a better manner, we designed three multihead fusion modules, and we chose the optimal model from the experiment. To ensure the accuracy of the experimental results, the parameters of the three models were the same. Model a directly contacts other scale features other than this layer, inputs them into the FA module to obtain new feature maps, and then inputs them into the prediction module to obtain the predicted results. Model b inputs the feature maps of different scales into the FA module and then concats the new feature maps. Unlike the first two models, model c first calculates V in Equation (5) corresponding to the different scale and layer features. The calculated result is subjected to a contact operation and is then added to the original scale feature map to obtain a merged feature map. It is shown in Table 3 that the performance of model c is optimal, and that the computational complexity is simpler compared to the other two models. Therefore, we choose model c as the multihead fusion module of MANet.

Table 3. Ablation study: effects of multiple models on the PASCAL VOC 2007 test.

Method	Network	Map
MANet (a)	ResNet–50	78.97
MANet (b)	ResNet–50	79.21
MANet (c)	ResNet–50	79.32

4.3.2. Impacts of Different Scales

To verify the validity of our multiscale fusion features added to the SSD, we specified different settings for the models and recorded their evaluated outcomes in Table 4. Based on the experiments, we gradually increased the features of the different scales for fusion (with the exception of the 1×1 and 3×3 scales, which were directly input to the final prediction module). We started with the large-scale features and gradually proceeded to merge them. From the presented results, it can be inferred that the performance impacts of the large-scale feature layer pairs were relatively large (the performance increases of the 38×38 and 19×19 scales were both increased by approximately 1%). We assume that this may be attributed to the fact that the underlying feature layer contained more information compared to the high-level features. At the same time, we can see that the integration of additional, different scale feature layers gradually improved the results. This proved that the fusion of different scale feature information was beneficial for the improvement of the detection performance that in turn justified the effectiveness of our model.

Table 4. Ablation study: effects of various scales on the PASCAL VOC 2007 test.

Convolution	Map (%)
—	76.67
38×38	77.00
$38 \times 38 + 19 \times 19$	78.61
$38 \times 38 + 19 \times 19 + 10 \times 10$	78.71
$38 \times 38 + 19 \times 19 + 10 \times 10 + 5 \times 5$	79.32

5. Conclusions

In this study, we presented a method for multiscale feature information fusion and demonstrated its effectiveness on a benchmark dataset. Compared with the Feature-fused SSD, we have combined more feature layers to achieve better results in accuracy. Compared with FSSD, we perform feature fusion operations on different scales. From the experimental results, our method further enhances the ability to express features. Additionally, our method still achieves better results. Compared with the ESSD method, our method is simply a feature fusion through the attention mechanism and does not introduce a visual reasoning mechanism. When the inputs size is 512, the accuracy of our method is better than ESSD. While we expected to find more efficient and effective ways to achieve improvements, experiments based on the use of the PASCAL VOC dataset have shown that the accuracy of our algorithm has greatly improved FPN. Our MANet model can surpass previous SSD frameworks, especially on small object. While we only applied our multiheaded attention model to the SSD framework, this approach can be applied to other detection methods. In the future, we will consider how to speed up our approach and we will try to replace the more powerful backbone network (such as DenseNet [34]) to improve the accuracy of detection. At the same time, we will verify that our bullish attention is equally valid for semantic segmentation tasks by replacing the FPN [6] in the Mask RCNN [35] with our fusion method.

Author Contributions: Conceptualization, J.J.; Data curation, Y.F.; Formal analysis, Y.F.; Methodology, H.X.; Project administration, S.Z.; Software, S.Z.; Writing–original draft, H.X.; Writing–review & editing, J.J.

Funding: This work was supported by the National Natural Science Foundation of China under Project 61873274.

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

References

1. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2015**, arXiv:1506.02640.
2. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
3. Ren, S.; He, K.; Girshick, R.; Jian, S. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv* **2015**, arXiv:1506.01497. [CrossRef] [PubMed]
4. 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, 2 July 2004; pp. 580–587.
5. Dai, J.; Yi, L.; He, K.; Jian, S. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *arXiv* **2016**, arXiv:1605.0640.
6. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Belongie, S. Feature Pyramid Networks for Object Detection. *arXiv* **2016**, arXiv:1612.03144.
7. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, 2999–3007. [CrossRef]
8. Everingham, M.; Eslami, S.M.A.; Gool, L.V.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [CrossRef]
9. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the International Conference on Computer Vision & Pattern Recognition (CVPR'05), San Diego, CA, USA, USA, 20–25 June 2005; pp. 886–893.
10. Felzenszwalb, P.F.; McAllester, D.A.; Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; p. 7.
11. Purkait, P.; Zhao, C.; Zach, C. Spp-net: Deep absolute pose regression with synthetic views. *arXiv* **2017**, arXiv:1712.0345.

12. Girshick, R. Fast R-Cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
13. 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]
14. Zitnick, C.L.; Dollár, P. Edge boxes: Locating object proposals from edges. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 391–405.
15. Carreira, J.; Sminchisescu, C. CPMC: Automatic object segmentation using constrained parametric min-cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1312–1328. [CrossRef] [PubMed]
16. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 354–370.
17. Jeong, J.; Park, H.; Kwak, N. Enhancement of SSD by concatenating feature maps for object detection. *arXiv* **2017**, arXiv:1705.09587.
18. Li, Z.; Zhou, F. FSSD: feature fusion single shot multibox detector. *arXiv* **2017**, arXiv:1712.00960.
19. Leng, J.; Liu, Y. An enhanced SSD with feature fusion and visual reasoning for object detection. *Neural Comput. Appl.* **2018**, 1–10. [CrossRef]
20. Fu, C.-Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. Dssd: Deconvolutional single shot detector. *arXiv* **2017**, arXiv:1701.06659.
21. Woo, S.; Hwang, S.; Kweon, I.S. Stairnet: Top-down semantic aggregation for accurate one shot detection. In Proceedings of the the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 1093–1102.
22. Cao, G.; Xie, X.; Yang, W.; Liao, Q.; Shi, G.; Wu, J. Feature-fused SSD: Fast detection for small objects. In Proceedings of the 9th International Conference on Graphic and Image Processing (ICGIP 2017), Qingdao, China, 14–16 October 2017; p. 106151E.
23. Luo, W.; Li, Y.; Urtasun, R.; Zemel, R. Understanding the effective receptive field in deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 4898–4906.
24. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
25. 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.
26. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
27. Cherry, J.M.; Adler, C.; Ball, C.; Chervitz, S.A.; Dwight, S.S.; Hester, E.T.; Jia, Y.; Juvik, G.; Roe, T.; Schroeder, M. SGD: Saccharomyces genome database. *Nucleic Acids Res.* **1998**, *26*, 73–79. [CrossRef] [PubMed]
28. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 Jun–1 July 2016; pp. 761–769.
29. Kong, T.; Yao, A.; Chen, Y.; Sun, F. Hypernet: Towards accurate region proposal generation and joint object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 845–853.
30. 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.
31. Tay, Y.; Tuan, L.A.; Hui, S.C. CoupleNet: Paying Attention to Couples with Coupled Attention for Relationship Recommendation. In Proceedings of the 12th International AAAI Conference on Web and Social Media, Palo Alto, CA, USA, 25–28 June 2018.
32. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
33. Shen, Z.; Liu, Z.; Li, J.; Jiang, Y.-G.; Chen, Y.; Xue, X. Dsod: Learning deeply supervised object detectors from scratch. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 Octtober 2017; pp. 1919–1927.

34. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Venice, Italy, 22–29 October 2017; pp. 4700–4708.
35. 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.