Journal of Visual Communication and Image Representation, Volume 78, July 2021, Article number 103189
DOI:10.1016/j.jvcir.2021.103189

Generative detect for occlusion object based on occlusion generation and feature completing

Can Xu[1], Peter Yuen[3], Wenxi Lang[4], Rui Xin[5], Kaichen Mao[1], Haiyan Jiang [1,2*]

1 College of Artificial Intelligence, Nanjing Agricultural University, Nanjing, 210095, Jiangsu, China

2 National Engineering & Technology Center for Information Agricultural, Nanjing Agricultural University, Nanjing, 210095, Jiangsu, China

3 Electro-Optics & Remote Sensing, Centre for Electronics Warfare, Information & Cyber (CEWIC), Cranfield University, Swindon, U.K

4. College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, Jiangsu, China

5. Department of Computer Science, Durham University, UK

Abstract: Detecting the object with external occlusion has always been a hot topic in computer version, while its accuracy is always limited due to the loss of original object information and increase of new occlusion noise. In this paper, we propose a occluded object detection algorithm named GC-FRCN (Generative feature completing Faster RCNN), which consists of the OSGM (Occlusion Sample Generation Module) and OSIM (Occlusion Sample Inpainting Module). Specifically, the OSGM mines and discards the feature points with high category response on the feature map to enhance the richness of occlusion scenes in the training data set. OSIM learns an implicit mapping relationship from occluded feature map to real feature map adversarially, which aims at improving feature quality by repair the noisy object feature. Extensive

experiments and ablation studies have been conducted on four different datasets. All the experiments demonstrate the GC-FRCN can effectively detect objects with local external occlusion and has good robustness for occlusion at different scales.

Keywords：Occlusion；Object detection；Feature completing；Generative Adversarial Networks；

1 Introduction

Object detection has always been an active field in computer vision research. Its goal is to learn a visual model for several kinds of objects and then use the model to predict the category and position of objects in the image. In recent years, thanks to the development of the convolutional neural network, related researches (Ren et al., 2016; Cai et al., 2016; He et al., 2017; Law et al., 2018; Lu et al., 2018; Zhou et al., 2019; Duan et al., 2019) on object detection have made a tremendous breakthrough in detection accuracy and speed, but the detection accuracy of objects which are in some complex scenes still needs to be further improved. The complexity is usually manifested by the presence of disturbing objects in the scene that are unrelated to the object to be detected. A typical example is that the detector may confuse trees with pedestrians at certain moments in the automatic drive. However, compared with the distinction between trees and pedestrians, the more difficult scene is to detect pedestrians blocked by trees, that is, to achieve accurate detection of blocked objects. For occluded objects, the loss of original object information and the mixing of irrelevant information increases the difficulty of feature learning. The low feature quality makes the detection results often contain a large number of False Negative samples. Therefore, how to realize the effective detection of the occluded objects has become the most important challenge of the detection algorithm in practical application.

Occlusion is a complex problem of optics and geometry. According to the causes, occlusion can be divided into two categories: intra-class occlusion and inter-class occlusion (Wang et al., 2018). In-class occlusion appears when the objects to be detected blocked by other objects in the same category, and studies have shown that (Ouyang et al., 2013; Tian et al., 2015) it mainly affects the positioning accuracy. That is, the detector can easily move the prediction box of object A to object B, which overlaps with object A. In recent work, Wang (Wang et al., 2018) designed a new constraint named Repulsion loss to promote each prediction box close to its ground truth box, while away from the ground truth box of other objects as far as possible. Zhang proposed a new detection algorithm named Occlusion Aware R-CNN, which designed the aggregation loss and PORoI to train several local detectors for the sub-area of the occluded object. By calculating the category probability and prediction frame coordinates, it finally fuses the results of every local detector, which improved the detection accuracy of the crowded pedestrians with the intra-class occlusion.

Here, we keep the point on the inter-class occlusion. Inter-class occlusion refers to the external occlusion caused by the coverage of different kinds of objects, whose difficulty lies in the poor feature representation of objects when detecting. Compared with conventional objects, it is harder to obtain high-quality features of inter-class occluded objects. Firstly, occlusion from other objects results in the loss of significant information of the object to be detected. On this basis, the features learnt cannot fully represent the object even if using the convolution neural network. Besides, occlusion means the original object data space will be mixed with noise. Furthermore, these local noises can be gradually transferred to the global feature with high-semantic information in the process of feature learning. Therefore, for the inter-class occlusion objects, how to achieve high-quality representation of object features is the key to improve the detection accuracy.

Different from Bell (Bell et al., 2016) and Lin (Lin et al., 2017) who fuse multiple convolution features to improve the feature quality of conventional objects, existing studies on occlusion detection (Pepik et al., 2013; Mathias et al., 2013; Tang et al., 2014; Gidaris et al., 2015; Zhou et al., 2017; Noh et al., 2018) pay more attention to mining the visible part. The core solution is: learning a series of local detectors for each part of the blocking object and using a specific strategy to fuse the results of local detectors to infer the final detection results of the whole object. Recently, Zhou (Zhou et al., 2017) proposed an occlusion detection method based on analyzing local occlusion and multi-label learning. By combining multiple local detectors, the correlation between local detectors is enhanced, which reduces the calculation cost and improves the detection accuracy of shielded objects. Noh (Noh et al., 2018) calculated the confidence of different regions of the occluded object and used the detection results of these visible regions to correct the final detection results of the whole object. Further analysis, we find that while exert visible region information fully may be effective to reduce the block noise, but to some extent also split the structure information between different parts, which caused a massive change on the results when combining different local regions to test. So, in this case, some specific prior knowledge of the occluded object is needed when designing the local detectors, which limit the generalization ability.

When it comes to improving the feature quality of the occluded object, the existing researches entirely mine the visible information of the unshaded area to suppress the occluded noise. Our solution is to complete the occlusion noise in the global feature map as we regard the inter-class occluded objects as the superposition of occlusion noise and original object information. We design a detection algorithm named GC-FRCN by introducing generative adversarial network to the Faster-RCNN [Ren et al., 2015], which mainly includes the OSGM and OSIM. The OSGM can simulate occlusion scenes by discarding the feature points with high category response on the feature map, aiming to construct training data that covers as many occlusion scenarios as possible to improve model's occlusion detection capability. The OSIM learns the implicit mapping relationship from occluded feature to real feature, and finally remove occlusion noise from the object's feature map. To ensure the mapping effectiveness, we make most use of the richer image information and constrain the mapping relation by keep the occluded images as similar as the real scene in both local details and global structure. The main contributions of this paper are as follows:

(1) We address the occluded object detection problem by expanding the richness of the occlusion scene and cleaning occlusion noise, and propose a cascading occlusion

detection algorithm GC-FRCN consisting of occlusion generation module OSGM and feature repair module OSIM. Experimental results on four different data sets demonstrate its superior performance.

(2) Different from the existing work, the simple yet effective OSGM discards the feature point with high category response and simulates different occluded scenes based on the analysis of effective receptive field. Our results show this strategy benefits the occlusion detection capability.

(3) With the implicit mapping relationship learnt by adversarially minimizing the difference between the occluded images and real scene in both local details and global structure, the OSIM can remove occlusion noise from the object's feature map. Our results show the OSIM has good robustness for occlusion at different scales.

2 Relate work

2.1 Generic Object Detection

Early researches on object detection relied on artificial features and classifiers to searching for the object to be detected in the image (Papageorgiou et al., 2000; Viola et al., 2004; Felzenszwalb et al., 2008; Felzenszwalb et al., 2009; Dollar et al., 2014). However, the detection accuracy is always unable to meet the actual application requirements, for the artificial features cannot express the object effectively. In recent years, due to the rapid development of convolution neural network, object detection algorithms based on deep learning have achieved breakthroughs in both detection accuracy and speed, which are mainly divided into two types: two-stage and single-stage object detection methods. Different from searching for regions of interest violently, the two-stage detection algorithm uses the generative strategy to produce proposals, which mainly includes RCNN (Girshick et al., 2014) and its subsequent improvements. RCNN automatically generates a set of candidate regions based on Selective Search algorithm (Uijlings et al., 2013), and then uses SVM and linear regression to achieve classification and position box fine-tuning, respectively. For its problem of extracting proposals' features repeatedly which cost a large of training resources, He (He et al., 2015) proposed a detection method based on spatial pyramid pooling which gets the proposals' features by mapping candidate regions on the global feature map; while Girshick (Girshick et al., 2015) directly trained an "end-to-end" CNN network to reduce the training volume. Furthermore, Faster RCNN (Ren et al., 2015) and R-FCN (Dai et al., 2016) combined the generation of candidate regions and detection of proposals into a whole network, which fine-tunes the entire network during training without storing a large number of features. Compared with the two-stage detection methods, the single-stage detection methods (Redmon et al., 2016; Liu et al., 2016; Redmon et al., 2017; Redmon et al., 2018) take the input image as a candidate region, and return object's boundary box coordinates and category on the preset anchor frames, which further improve the training efficiency and detection speed of the detector.

2.2 Data Augmentation

Sufficient training data is the foundation for constructing a deep learning model. The CNN gradually abstracts the features from the original images, so the quality and quantity of training images have a direct effect on features' effectiveness. As a result, the performance of the detector will generally improve with the increase of the scenes containing objects in training data. However, collecting and making an extensive detection data set is so difficult that the usual treatment is to expand and enhance the available training data through operations such as rollover, rotation, scaling, clipping and shifting. Meanwhile, some studies (Simo et al., 2014; Loshchilov et al., 2015; Wang et al., 2015) also explored how to fully mine and utilize the limited training data to improve the accuracy and robustness of the detector. Shrivastava (Shrivastava et al., 2016) proposed a detection method based on difficult sample mining, which significantly improved the detection accuracy by retraining samples with massive losses. Wang (Wang et al., 2017) also showed that the detector's robustness on shielding and deformation could be improved by continuously constructing shielding and deformation samples when training the detector. In this paper, we are also inspired by data enhancement to generate a large number of occlusion samples to enhance the diversity of training data and further improve the detection performance of the model for inter-class occlusion objects.

2.3 Feature Completing

For inter-class occluded objects, we hope to restore noise in the features as the real information partially lost due to occlusion, to improve the feature quality as well as detection accuracy. Although the research of feature repairing is still in the initial stage, the problem of image repairing has been widely studied. The purpose of image repairing is to automatically recover the lost content in the image, whose early methods focus on repairing by spreading the known local information to the unknown.

With the breakthrough of the generative adversarial network in the application of image repairing, relevant researches (Xiang et al., 2017; Lahiri et al., 2017; Yeh et al., 2017; Dolhansky et al., 2018) have achieved more accurate results not only in semantic but also the visual effect of repairing details. Recent studies expand the structure of the generative adversarial network by using multiple discriminators to improve the repairing effect further. Pathak (Pathak et al., 2016) proposed an encode-decode network for image repairing; and then Iizuka (Iizuka et al., 2017) designed the repair network based on local and global discrimination models, which realized the optimization of local details and overall texture of the image. On this basis, Li (Li et al., 2017) further added the semantic parsing model to optimize the face structure information, which reduces the error to the human eye level. Yu (Yu et al., 2018) abstracted the repair process into two encode-decode steps and further optimized the repair results with coarse precision by using counter loss, which significantly improved the repair accuracy.

3 Generative Features Completing

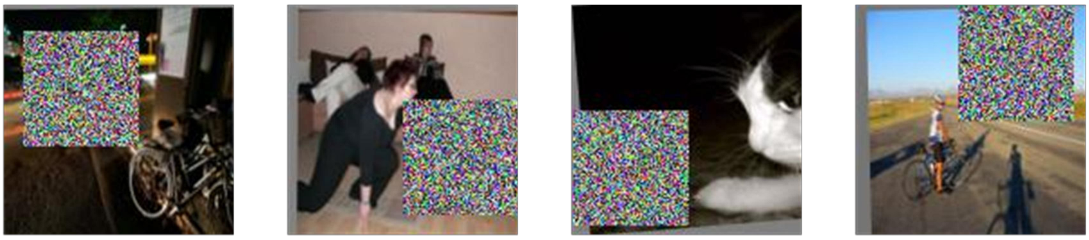
Based on the data-driven strategy, we improved the feature quality and constructed the occlusion object detector by expanding the richness of the occlusion scene and cleaning occlusion noise in the feature. Here, the key is how to generate representative occlusion data and repair occlusion noise, for which we designed OSGM and OSIM, respectively.

3.1 OSGM: Refinement for the Occlusion Generation

3.1.1 Analysis of Occlusion Simulation

Deep learning methods are always based on large-scale data learning to achieve the abstraction and modelling of a certain type of problem. When detecting objects with local occlusion, the simple solution is to construct a data set covering all occlusion scenarios. However, collecting a sufficiently large occluded data set is complicated and low cost-effective. Without extra data collection work, a feasible occlusion simulation method is to randomly discard pixels of different combinations on the existing detection data set. However, it cannot guarantee the effectiveness and representativeness of occlusion scenes. As shown in Fig. 1, objects are not blocked in some images because positions of objects to be detected and the pixels to be discarded are random. With the decrease of the discarded size, the number of similar invalid samples will further increase substantially. For the same object, there will be much redundancy when simulating different occlusion scenes, whose occlusion expression may be more similar after feature learning. Invalid samples and repeated samples do not help improve the performance of the model but bring additional computational overhead for feature learning and subsequent repair.

a) Repeated occlusion scenarios



b) Invalid occlusion scenarios

Fig. 1 Examples of invalid and repeated occlusion scenarios

3.1.2 Design of Occlusion Simulation

Based on the analysis in section 3.1.1, we hope that occluded image generated not only represent a kind of occlusion scene, but also the object is always blocked. For this reason, we firstly discard pixels on the feature map to ensure enough differences of different occlusion scenes generated based on the same object. During feature learning, the original input image will be abstracted into the feature map iteratively, and the pixels on the feature map have more robust semantics than the original image. Different feature maps after discarding pixels can be approximated as the abstraction of different occlusion scenes. The area of the image that any pixel of the feature map corresponding to can be described as a theoretical receptive field. When generating occluded samples, what we need to drop out is these pixels in the theoretical receptive field of the input image. For a specific network, the calculation method of the theoretical receptive field is shown in formula (1).

$$S_{RF}(t) = (S_{RF}(t-1) - 1)N_s(t) + S_f(t) \tag{1}$$

Where the $S_{RF}(t)$ means the theoretical field size of convolution layer t , while $N_s(t)$ and $S_f(t)$ is the stride and convolution kernel size of convolution layer t.

In order to eliminate invalid occlusion scenes, we also want to discard pixels that are highly relevant to the object. Luo (Luo et al., 2016) found that although the value of pixel on the feature map is determined by the value in the receptive field of image, the correlation degree between different image pixels and feature map pixels is quite different. Compared with the pixels at the edge of the image, the pixels in the middle of the image have more influence on the value of feature map, and the effective receptive field which actually decides the value of feature map is always smaller than the theoretical receptive field. In other words, compared with the edge, the pixels in the middle of feature map are affected by more original image information during the convolution calculation, which means a higher probability to contain the original information of the object. We chose to discard the pixels in the middle of the feature map which are more relevant to the target to be detected. For the $N \times N$ feature map, if the pixel coordinates of its upper left vertex are denoted as (x_0, y_0), the range of disposable pixel coordinates (X_{erf}, Y_{erf}) can be calculated by formula (2)-(4).

$$\begin{aligned} X_{erf} &\in (x_0 + \lceil \alpha * N \rceil,\ \ x_0 + \lfloor (1-\alpha) * N \rfloor) \\ Y_{erf} &\in (y_0 + \lceil \beta * N \rceil,\ \ y_0 + \lfloor (1-\beta) * N \rfloor) \end{aligned} \tag{2}$$

$$\alpha = \frac{w_{obj}}{w_{in}} \tag{3}$$

$$\beta = \frac{h_{obj}}{h_{in}} \tag{4}$$

Where α and β represents the significant discard coefficient; w_{obj} and h_{obj} represents the width and length of the object's minimum enclosing rectangle; w_{in} and h_{in} means the width and length of the input image, respectively.

3.1.3 Structure of OSGM

As shown in Fig. 2, the basic structure of OSGM is from the conv1 layer to the pool3 layer of VGG16 network. For all the convolution layers, we adopt the kernel of 3×3 and add standard Batch-Normalization and Relu operation. While for the pooling layers, we use max pooling with a kernel of 2×2. OSGM determines the pixels' effective discard range of feature map using the formula (2) - (4) and calculates the receptive field using the formula (1). Then, we set the values of all pixels as 0 in the corresponding to the receptive field, which is mapped by the pixel drop out from the feature map. Here, we directly reuse the VGG16 model trained on the ImageNet data set to initialize the parameters of OSGM. Besides, in order to further enhance the richness and difficulty of occluded samples, we designed four different occlusion templates with the size of 1×1, 1×2, 2×1 and 2×2 when discarding pixel points in the feature map.

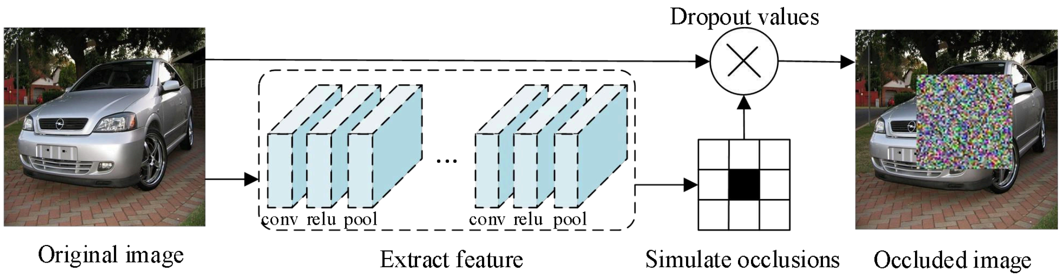


Fig. 2 Structure and workflow of the OSGM module

3.2 OSIM: Refinement for the Occlusion Representation

3.2.1 Overview of Occlusion Inpainting

For the object with local occlusion, the occlusion noise mixed with the original data space will run through the feature learning, resulting an upper limit of detection accuracy. Our innovative idea is to learn an implicit mapping relationship from occluded feature map to real feature map. To realize this goal, as shown in Fig. 3, OSIM is composed of one Generator and two discriminators, which make the repaired region consistent with the real label both in local details and overall structure.

3.2.2 Generator

The generator is described as a process of feature learning and generating new feature values for the occlusion region. As shown in Fig. 3, after the generator learning the object features based on the encoding, it generates new feature values for the occlusion object and then passes them to the discriminator. The encoding network is based on the conv1 to pool2 layers of the VGG16 network (Simonyan et al., 2014), where the convolution kernel is 3×3 and the max pooling kernel is 2×2. We use L_2 loss to measure the difference between generated features and real features. The L_2 loss function of the generator is shown in formula (5).

$$L_G = \frac{1}{2M} \sum_{i=1}^{M} \left\| x_i - x_i^{'} \right\|_2^2 \tag{5}$$

Where M is the number of pixels on the feature map, x_i and $x_i^{'}$ means the real and the generated feature pixels.

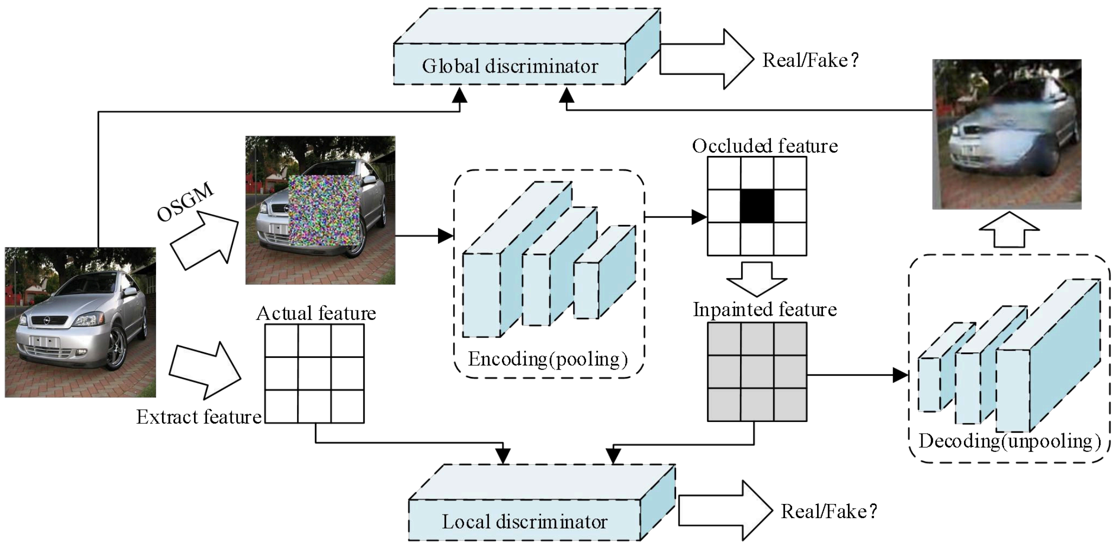


Fig. 3 Structure and workflow of OSIM module

3.2.3 Discriminator

The generator makes a narrow gap between the feature values containing block noise and its corresponding real values, but it cannot guarantee the repaired features similar to the real features in terms of content and distribution. It is because the L_2 loss punishes the outliers seriously and does not consider the local context information and structural relationship between the occluded region and its adjacent region. Ideally, the restored features should be not only similar to the real features in content, but also be similar to the surrounding regions in structure. For this reason, we designed the local discriminator and global discriminator, respectively in OSIM to constraint the features generated further. As shown in Fig. 3, the local discriminator focuses the attention of the generator on the internal details of the occlusion region, which helps the repaired features to be consistent with the real features in terms of pixel value and statistical distribution. The global discriminator maps the restored features to the same size as the input image through the decoding network, which normalizes the structural relationship by identifying the similarity between the original input image and the image upsampled from the repaired feature map. It should be noted that, the structure of the encoding network and the decoding network is symmetrical, while the only difference between the two networks is that the un-pooling layer is used to replace the pooling layer in the decoding network.

We also note that the network structure of the local discriminator and the global discriminator is similar to the research proposed by Radford (Radford et al., 2016). Furthermore, the two discriminators also have the same loss function which is shown in formula (6).

$$L_{localD} = L_{globalD} = \min_{G}\max_{D} E_{x\sim P_{data}(x)}[logD(x)] + E_{z\sim P_z(z)}[\log(1 - D(G(z)))] \tag{6}$$

Where L_{localD} and $L_{globalD}$ represents the loss function of local discriminator and global discriminator, $E_{y\sim P_{data}(y)}$ and $E_{z\sim P_{z}(z)}$ represents the distribution of the true image pixels and occluded noise. The loss function L of OSIM module consists of generator and discriminator which can be calculated by formula (7).

$$L = L_G + \gamma_1 L_{localD} + \gamma_2 L_{globalD} \tag{7}$$

Where γ_1 and γ_2 are used to balance the loss of different parts, and the default value is both 300.

4 GC-FRCN：Approach Details

4.1 Structure of GC-FRCN

As shown in Fig. 4, GC-FRCN takes Faster-RCNN as the basic network structure, and includes five key steps: occluded data generation based on OSGM, feature learning, repairing feature based on OSIM, candidate region generation, object classification and position box regression. To ensure the reuse of occluded data set generated, OSGM is designed as an independent module which cascade integrated into GC-FRCN. For the different occluded data generated by OSGM, GC-FRCN uses the convolution neural network to learn the global features of the whole image and outputs the feature map. Here, the critical role of OSIM is to provide more accurate feature representation of blocked objects, so the OSIM is embedded as a plug-in after the feature learning step which is trained independently and transmits the repaired feature map to the RPN (Region proposal network, RPN). RPN uses the sliding window to traverse the repaired feature map, and sets 9 rectangular regions (3 aspect ratios × 3 scales) to generate candidate regions when mapping each pixel of the feature map. Finally, the restored features are maximally pooled to obtain the features of each candidate regions, which are fed into a cascade of entirely complex networks to achieve the final category and position box.

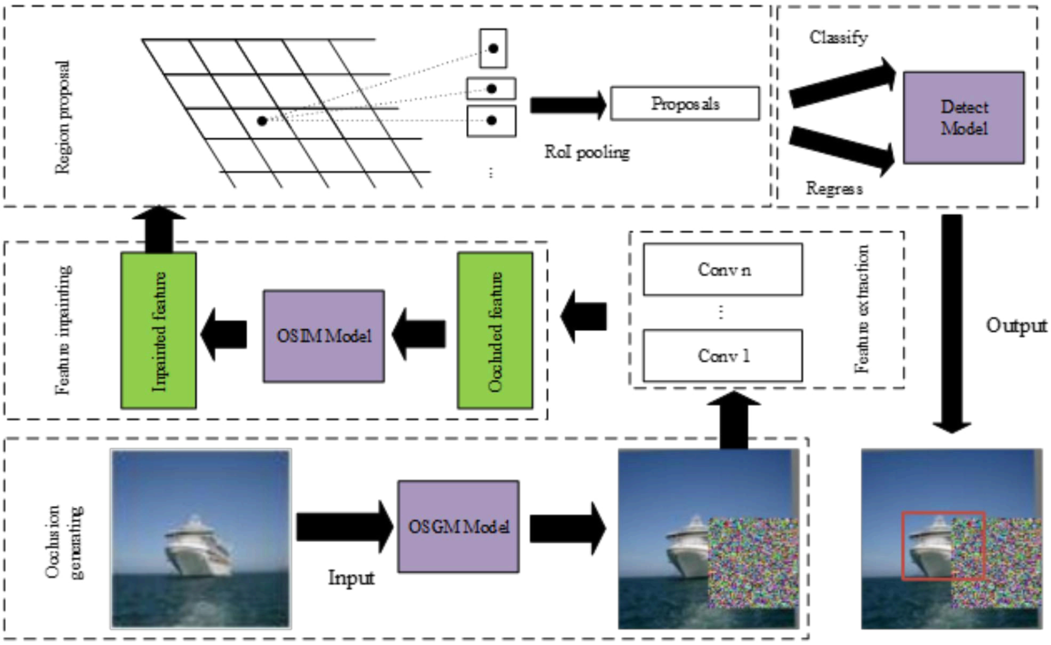


Fig. 4 Structure and workflow of GC-FRCN module

4.2 Independent Training for GC-FRCN

In this study, the training of GC-FRCN includes two parts: training the OSIM and training the detector. When training the repair model OSIM, the generation loss L_G is used firstly to fill the initial eigenvalue for the occluded object; and then the discriminator loss L_D is used to improve the precision of the eigenvalue. We initialize the parameters of OSIM randomly at the beginning of training, but the model of the latter stage is trained based on the model obtained from the previous stage in order to improve the training efficiency and model accuracy. When training the detector, we follow the setup of standard Faster RCNN based on SGD (Stochastic gradient descent, SGD) and alternate optimization strategy, where the only difference is the feature passed to RPN optimized by the repair model in the first place. The loss function of the detector is composed of classification loss and regression loss, which are normalized by N_{cls} and N_{reg} and then weighted by equilibrium parameters λ (). The loss function is shown in formula (8).

$$L(\{P_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(P_i, P_i^*) + \lambda\frac{1}{N_{reg}}\sum_i P_i^* L_{reg}(t_i, t_i^*) \quad (8)$$

Here, N_{cls} represents the mini-batch size of training, N_{reg} represents the number of candidate regions and the i is the anchor number. P_i is the probability of the anchor point being as an object, and the corresponding P_i^* value is given as 1 when the anchor point is predicted as positive and otherwise it is 0 if the anchor is negative. t_i and t_i^* represent the coordinates of the upper left and lower right vertex of the predicted bouncing box respectively. The L_{cls} and L_{reg} can be calculated by formula (9) and (10).

$$L_{cls}(P_i, P_i^*) = -\log[P_i^* P_i + (1 - P_i^*)(1 - P_i)] \quad (9)$$

$$L_{reg}(t_i, t_i^*) = \begin{cases} 0.5(t_i - t_i^*)^2 & |t_i - t_i^*| < 1 \\ |t_i - t_i^*| - 0.5 & |t_i - t_i^*| \geq 1 \end{cases} \quad (10)$$

5 Experiment

5.1 Datasets and Evaluation Metrics

To verify the performance of GC-RFCN, we had carried out several experiments on four data sets of PASCAL VOC 2007, VOC 2012 (Everingham et al., 2010), MS COCO (Lin et al., 2014) and PANICLE2017. The PANICLE2017 is an image data set containing rice panicles covered by leaves. As shown in Fig. 5, PANICLE2017 consists of two parts. The first one is marked according to the format of VOC, which is used to train the rice panicle detector. The training data set, verification set and test set are composed of 2080, 912 and 1280 field rice images, respectively. The other part is composed of 982 images of unshaded rice panicles, which are used to train the occlusion feature repair model.

We conducted most of the ablation studies on the PASCAL VOC 2007 data set and the COCO data set and reported the results of verification of the actual application effect on the PANICLE2017 data set. First, we select the mean average precision (mAP) and mean average recall (mAR) to evaluate the performance of GC-FRCN on VOC and COCO data sets, as shown in formula (11) and (12).

$$\mathrm{mAP} = \frac{1}{m}\sum_{i=1}^{n} P_i \left(R_i - R_{i-1}\right) \tag{11}$$

$$\mathrm{mAR} = \frac{1}{m}\sum_{i=1}^{n} 2\int_{0.5}^{1} R_{IoU} d(IoU) \tag{12}$$

Where R_i represents the different recalls ranked according to the confidence degree, and P_i represents the maximum precision corresponding to the R_i. And the R_{IoU} means the recall corresponding to the IoU (Intersection-over-Union, IoU). Secondly, in order to estimate the restoration accuracy of OSIM quantitatively, SSIM (structural similarity index) was selected to evaluate the difference of images before and after image restoration, which can be calculated in formula (13).

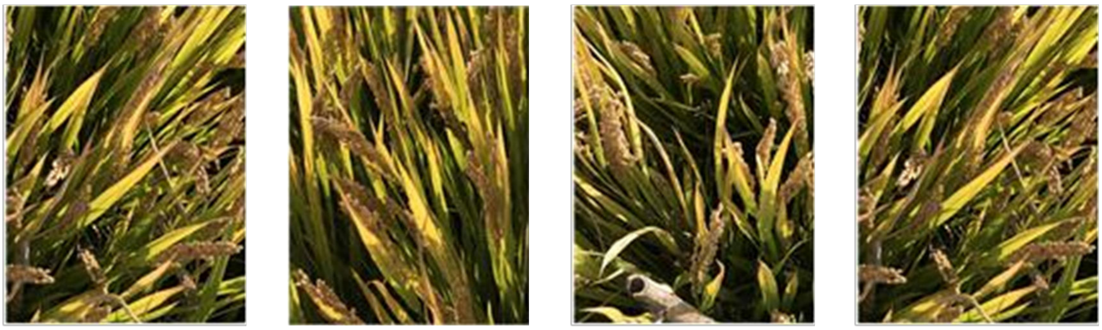
$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(\sigma_{xy} + c_2)}{({\mu_x}^2 + {\mu_y}^2 + c_1)({\sigma_x}^2 + {\sigma_y}^2 + c_2)} \tag{13}$$

Where x and y represents the original image and recovered image; μ and σ represents the average and the standard deviation of x and y, while the σ_{xy} means the covariance of x and y; The c_1 and c_2 are constants to avoid the denominator being 0 whose default value are 6.5025 and 58.5225, respectively. Thirdly, we select the counting accuracy and the classification accuracy to evaluate the performance of GC-FRCN on PANICLE2017 data set. The counting accuracy P_c refers to the ratio of detecting the correct number of panicles to the actual number of panicles; while the classification accuracy P_t is the correct number of panicles identified as panicles (true positive) to the number of all objects identified as panicles (true positive and false positive) in the imagery data set:

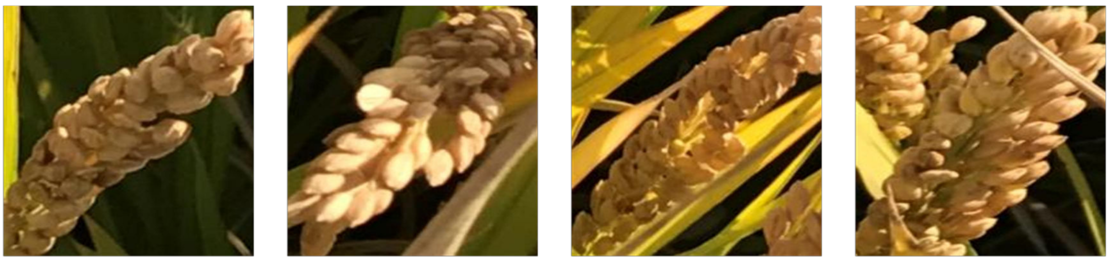
$$P_c = \frac{N_{cor}}{N_{real}} \times 100\% \tag{13}$$

$$P_t = 1 - \frac{N_{err}}{N_{dect}} \times 100\% \tag{14}$$

Where N_{cor} and N_{err} are the correct (true positive) and wrong (false positive) number of panicles detected by the model, respectively; N_{real} and N_{dect} represents the actual number of panicles and all the objects identified as panicles in the test sample.



(a) Training images of VOC2017 dataset for the detect model



(b) Training images of VOC2017 dataset for the detect model

Fig. 5 Training images of PANICLE2017 rice data set

5.2 Experiment Settings

As described in section 3.1, all experiments were simulated by OSGM module to reconstruct the experimental data set. For VOC data sets, we used 'trainval' set and ‘test’ set for training and testing, respectively. For the feature repair model, we used a 250K SGD training generator and discriminator by keeping the learning rates at 0.0001 and 0.0002, respectively. For the detector, the number of iterations is 80 k and the learning rate starts from 0.001 and decreases to 0.0001 after 60K iterations. Also, we followed most of the training setups of the standard Faster RCNN (Ren et al., 2015) with a mini batch size of 2 images and candidate regions of 256. For the COCO data set, we used ‘trainval35k’ set and ‘minival’ set for training and testing, respectively. The parameters of feature repair model are the same as those of VOC data set. For the detector, the number of iterations is 320K, and the initial learning rate is 0.001, which decreases to 0.0001 after 280K iterations. For the PANICLE2017 data set, the feature repair model and detector will keep all parameter settings consistent with the VOC data set.

When test the model, the experimental results of PANICLE2017 data set were obtained from the test set composed of real field scenes. For the VOC data set and the COCO data set, we generate occlusion at different scales (small, medium and large) on the ‘test’ set of VOC and ‘minival’ set of COCO using four discard the template (1×1, 1×2, 2×1 and 2×2). Especially, the small, medium and large scale mean the about 6%, 14% and 25% pixel loss of the whole image respectively, while means the 14%~22%, 20%~31% and 46%~60% pixel loss of the object to be detected.

5.3 Results on PASCAL VOC 2007

5.3.1 Quantitative Evaluations of GC-FRCN

In order to verify the effectiveness of GC-FRCN, we select the classical Faster

RCNN as the baseline and combine our OSGM and OSIM to train detectors, respectively. The results are shown in Table 1. Taking small scale occlusion and ZF network as an example, mAP of baseline is 48.6%, which has an increase of 2.8% and 3.3% after adding OSGM module and OSIM respectively. While the static Faster-RCNN with ZF-net achieves a mAP of 58.7% on the VOC 2007 test without occlusion, which is about 32% and 10% higher than the big occlusion and small occlusion. From this point of view, we can find the occlusion has a significant effect on the detect results, and the difficulty of repairing the detecting is increasing with the size of occlusions. All these rising trends are also reflected in the test results of large and medium scale occlusion.

Table 1 Mean average precision for **VOC 2007 test** with different size of occlusions

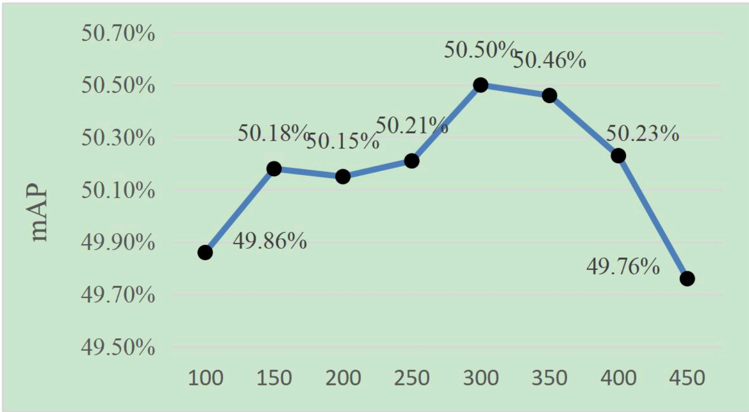
Method	Arch	Mechanism		mAP of different occlusion(%)			
		+OSGM	+OSIM	Big	Middle	Small	None
Faster-RCNN(Baseline)	VGG16			36.9%	53.5%	59.3%	66.9%
	VGG16	✓		43.9%	55.1%	61.5%	/
	VGG16		✓	48.1%	59.4%	63.5%	/
	ZF			26.7%	41.7%	48.6%	58.7%
	ZF	✓		33.9%	41.1%	51.4%	/
	ZF		✓	38.9%	48.6%	52.9%	/
A-FRCN	VGG16			45.2%	53.7%	60.6%	69.1%
YOLO	VGG16			43.6%	52.7%	58.8%	65.8%
YOLO V3	VGG16			47.3%	58.6%	63.6%	76.3%
SSD	VGG16			46.4%	58.1%	62.9%	72.2%
GC-FRCN (Ours)	VGG16	✓	✓	50.5%	61.1%	65.1%	69.9%

We compare our method with other state-of-the-art detection methods on the backbone of VGG16. The mAP of baseline for the large, medium and small scale occlusion are 36.9%, 53.5% and 59.3%, which increase significantly after introducing the OSGM and the OSIM further. For our GC-FRCN, the mAP of 50.5%, 61.1% and 65.1% for three occlusion scales, outperforming baseline by 13.6%, 7.6% and 5.8%.

Furthermore, among the purely one-stage detectors such as YOLO, YOLO V3 and SSD or the two-stage like A-FRCN (Wang et al., 2017), the best result of YOLO V3 is 47.3% for large occlusion scale while 63.6% for small occlusion scale, which are lower by 3.2% and 1.5% than the GC-FRCN. The comparison results on the PASCAL VOC 2007 are presented in Table 1. The results show that GC-FRCN can effectively improve the detection accuracy of objects with different occlusion scales.

5.3.2 Ablative Analysis

Hyper-parameter Analysis. The γ_1 and γ_2 in formula (7) determine the influence of the generator and discriminators on the occlusion impainting task, which is the key hyper-parameters in our OSIM. To find their optimal values, we conduct experiments using the OSIM model training from different γ_1 and γ_2**.** We always set same value for γ_1 and γ_2. Intuitively, it may make more sense to find out the relationship between our generator and discriminators due to the two discriminators working as a whole participate in the zero-sum game with the generator. As shown in Fig. 6, the detection performance (reported by mAP of big occlusion scale) can be obviously improved by setting the $\gamma_1 = \gamma_2 = 300$. We suppose the too small weight is difficult to contribute the key feature generation, and too large weight means too harsh on the generator and may result in a local optimal solution.



Different γ_1 and γ_2 for OSIM model

Fig. 6 The selection weights for local and global discriminator loss

OSGM Analysis. As shown in Table 2, to verify the effectiveness of OSGM, we also compared it with other occlusion generation strategies. We used the occlusion simulation strategy of discarding pixel values randomly on the original image as the benchmark. At this time, take the small scale occlusion as an example, the mAP as well as mAR of objects are 65.5% and 78.9%, and the model training time is about 610 minutes. The second strategy is to randomly discard pixels on the feature, whose result shows that discarding pixels from the feature map is equivalent to discarding original pixel values directly. When it comes to our OSGM which selects and discard high-semantic feature points, the mAP and mAR only decreases by 0.3% and 0.5% compared with the second strategy. We also find the training time has a dramatic reduction in our OSGM, which decreases by nearly 33% in contrast to the second strategy and decreases by more than 50% from baseline. We suppose that our OSGM can significantly reduce the training cost during screen and produce high representative and effective occlusion scenes.

Table 2 Results of GC-FRCN for **VOC 2007 test** with different OSIM drop strategies

Methods	mAP of different occlusion			mAR of different occlusion			Training time
	Big	Middle	Small	Big	Middle	Small	
Drop on image	50.6%	61.4%	65.5%	61.7%	73.1%	78.9%	610min
Drop on feature map	50.6%	61.3%	65.4%	61.7%	72.8%	78.7%	415min
Drop on Effective RF (OSGM)	50.5%	61.1%	65.1%	62.2%	72.5%	78.4%	275min

OSIM Analysis. We also use different loss function to train repair models and then compare the detection accuracy of GC-FRCN for occlusion at different scales. The simplest baseline method is to train the repair model using only the generation loss L_G, as shown in the first row of Table 3, whose mAP and mAR for the object with small scale occlusion is 62.6% and 75.7%. In another set of experiments, we add local discrimination loss L_{localD} to train the feature repair model, at which time the mAP and mAR for small-scale occluded object increases by 1.4% and 0.8%. When the loss function L_3 is used to normalize the feature repair model, we show the optimized occluded object which output by the global discriminator in Fig. 7. The visualization results show that the OSIM structure in this paper can effectively remove the occlusion noise in the feature. We obtain a mAP of 65.1% for the small-scale occluded object, which increases by 2.5% and 1.1% in contrast to L_1 and L_2 respectively. Similarly, for the object with large or medium scale occlusion, the detection accuracy of GC-FRCN still increases with the refinement of the repair network structure and loss function. All the experimental results show that our OSIM can improve the repair accuracy of the features and further improve the detection accuracy of GC-FRCN.

Table 3 Results of GC-FRCN for **VOC 2007 test** with different OSIM loss functions

Different loss	Mechanism			mAP of different occlusion			mAR of different occlusion		
	$+L_G$	$+L_{localD}$	$+L_{globalD}$	Big	Middle	Small	Big	Middle	Small
L_1	✓			42.4%	56.5%	62.6%	54.9%	69.1%	75.7%
L_2	✓	✓		44.8%	57.8%	64.0%	57.3%	69.8%	76.5%
L_3	✓	✓	✓	50.5%	61.1%	65.1%	61.6%	72.7%	76.5%

In addition to the mAP of the detection, we also perform a quantitative evaluation using the three loss functions on the three different occlusion scales. The results are shown in Talbe4. For the first row, we can see the SSIM is 0.703 for the small occlusion scale while only fall by 3.2% for the big occlusion. Comparing to the results of the second and third row with the discriminators, the SSIM of L_1 shows a better stability with the change of occlusion. We suppose this is because the L_1 favors more on the distance in pixel values simply. In other words, the L_1 performs poorly as it hardly recovers the useful semantics to some extent, which can explain the lower mAP in Table 3. After adding discriminators, OSIM with the L_3 achieves a SSIM of 0.728 for the big occlusion, which increases by 10.4% compared to the SSIM of 0.804 for the small occlusion. At the same time, we also find all SSIM of our OSIM with the L_3 are better than the L_1 and L_2. These gaps between different occlusion scales and different loss functions show the validity and rationality of our OSIM with two discriminators.

Table 4 SSIM of OSIM for **VOC 2007 test** with different loss functions

Different loss	Mechanism			SSIM of different occlusion		
	$+L_G$	$+L_{localD}$	$+L_{globalD}$	Big	Middle	Small
L_1	✓			0.671	0.686	0.703
L_2	✓	✓		0.695	0.731	0.746
L_3	✓	✓	✓	0.728	0.773	0.804

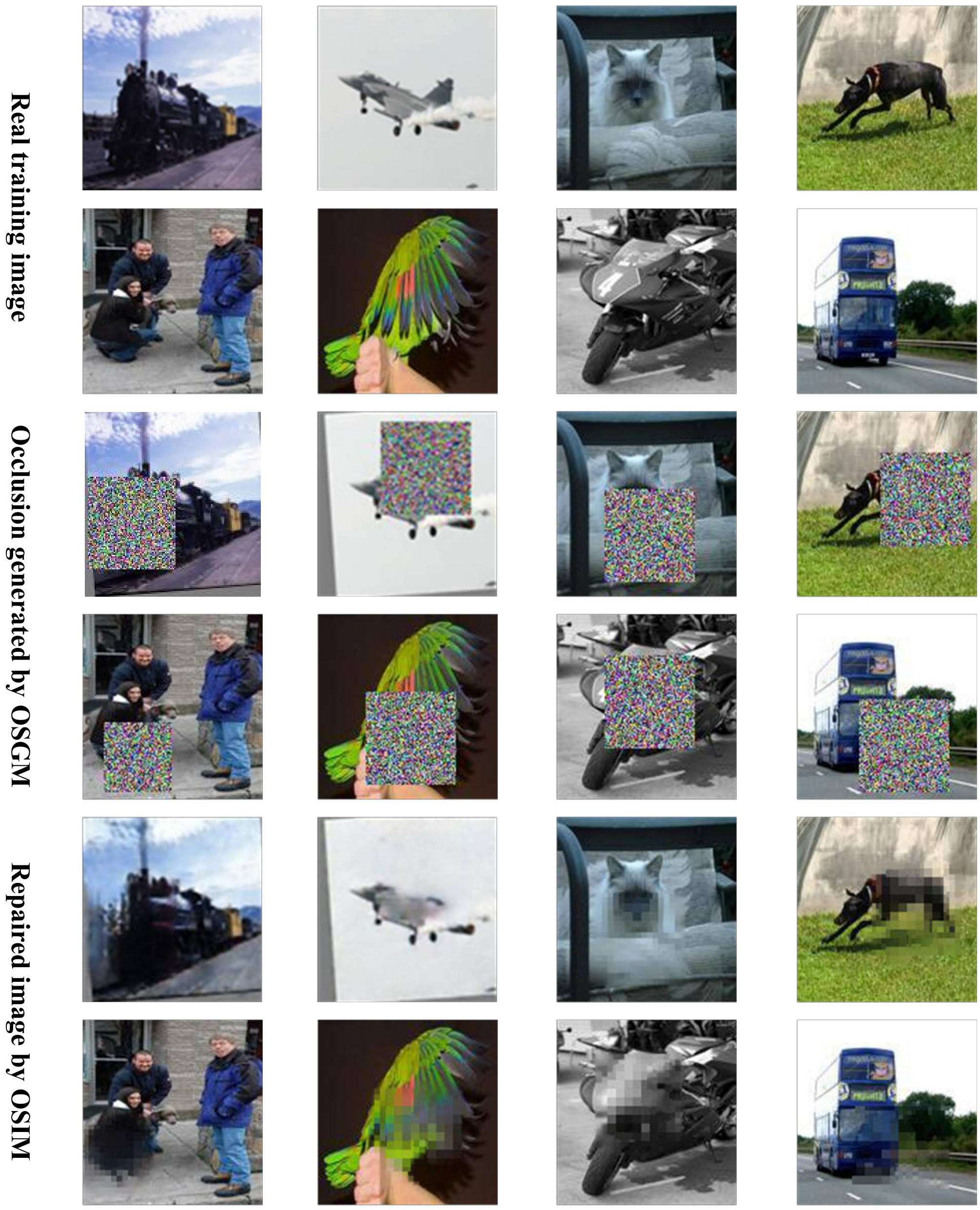


Fig. 7 Visual repair result based on OSIM

5.3.3 Category-based Analysis

Table 5 shows the change of detection accuracy of GC-FRCN and Faster RCNN for different categories of objects under different scales of occlusions. Firstly, the detection accuracy of GC-FRCN varies significantly for different kinds of objects. Taking 'bicycle' and 'train' as examples, the AP of the three kinds of occlusion scales are all over 59%, which can reach 76.5% and 74.6% respectively for the small-scale occlusion. However, for 'bottle' and 'potted plant' with small scale occlusion, the APs are only 40.1% and 34.9% respectively and will continue to decrease with the increase of occlusion scale. Moreover, we also find though our GC-FRCN can effectively improve the detection accuracy of most categories under different occlusion scales, the improvement of GC-FRCN is not obvious for some categories and even decrease slightly compared with the baseline method in some cases. More interestingly, these cases also mainly focus on the ‘bottle’ and ‘pottedplant’. The possible explanation is that compared with other objects, it is more difficult to learn features of small-size objects such as the ‘bottle’ and ‘potted plant’, which also makes it more difficult to deduce occlusion when using original real features.

Table 5 Changes of average precision of GC-FRCN relative to baseline for **VOC 2007 test**

Category	AP of GC-FRCN			AP of Faster-RCNN			Change of AP		
	Big	Middle	Small	Big	Middle	Small	Big	Middle	Small
aeroplane	49.3%	58.0%	66.6%	23.4%	42.4%	48.5%	26.0%	15.6%	18.1%
bicycle	63.1%	75.1%	76.5%	46.0%	69.0%	78.0%	17.1%	6.1%	-1.6%
bird	36.6%	54.1%	61.8%	30.3%	50.7%	53.5%	6.3%	3.4%	8.3%
boat	40.7%	47.7%	53.2%	27.3%	42.6%	43.4%	13.4%	5.1%	9.8%
bottle	31.1%	38.2%	40.1%	30.2%	41.8%	42.5%	0.9%	-3.6%	-2.4%
bus	63.0%	73.7%	72.3%	41.8%	61.8%	68.1%	21.3%	11.9%	4.2%
car	65.8%	74.2%	75.7%	52.7%	68.4%	75.1%	13.1%	5.8%	0.6%
cat	63.6%	74.4%	77.3%	40.8%	61.8%	69.0%	22.8%	12.6%	8.3%
chair	33.9%	43.7%	46.6%	22.3%	40.3%	42.9%	11.7%	3.4%	3.7%
cow	43.8%	62.5%	66.5%	38.7%	54.4%	61.8%	5.1%	8.1%	4.7%
diningtable	57.3%	67.2%	66.0%	42.2%	62.7%	61.6%	15.1%	4.6%	4.4%
dog	58.1%	70.7%	73.6%	37.4%	56.3%	64.9%	20.7%	14.3%	8.7%
horse	65.1%	73.7%	77.7%	52.2%	68.7%	75.4%	12.8%	4.9%	2.3%
motorbike	62.0%	69.9%	72.8%	43.2%	64.1%	66.4%	18.8%	5.7%	6.4%
person	54.9%	63.9%	68.5%	45.4%	56.7%	66.7%	9.5%	7.2%	1.8%
pottedplant	28.5%	32.9%	34.9%	28.8%	30.9%	35.4%	-0.3%	2.0%	-0.5%
sheep	33.9%	54.8%	64.6%	30.7%	51.1%	57.0%	3.1%	3.8%	7.6%
sofa	52.1%	61.7%	63.7%	33.2%	54.5%	58.6%	18.9%	7.1%	5.1%
train	59.0%	67.3%	74.6%	35.8%	49.6%	59.7%	23.2%	17.8%	14.9%
tvmonitor	49.1%	58.6%	68.5%	34.3%	42.5%	57.8%	14.8%	16.1%	10.8%

5.3.4 Different Size of Occlusion Analysis

According to the results in table 1, for objects with large, medium and small scale occlusion, the mAP of baseline is 36.9%, 53.5% and 59.3% respectively, which increase with the decrease of the occlusion scale. Other experimental results in table 1 also verify this trend of accuracy change. For GC-FRCN, the mAP of objects in small scale occlusion is 65.1%, which is significantly increased by about 15% than that of objects in large scale occlusion. The change of occlusion scale directly describes the amount of occlusion noise and the loss degree of the original information. The above experimental results support the hypothesis that occlusion noise will directly affect the classification accuracy in this paper.

Secondly, for the OSGM and OSIM modules involved in GC-FRCN, we find that there are significant differences in the improvement of the accuracy of objects with different occlusion scales. Compared with the baseline, for the objects with three different occlusion scale, the detection accuracy increases by 2.2%, 1.6% and 7% after adding the OSGM module, while increases by 4.2%, 4.7% and 11.2% after adding the OSIM module respectively. The above experimental results show that compared with OSGM module based on data enhancement strategy, OSIM based on high-quality feature expression strategy has a more noticeable improvement in the detection accuracy of occluded objects. Besides, the improvement of detection accuracy of OSIM is more and more evident with the increase of occlusion scales. When analyzing this phenomenon in-depth, the reason may be the lack of available original effective information for the repairing of objects with large scale occlusion, which increases the difficulty of repairing and reduces the detection accuracy; In contrast, compared with the small scale occlusion object which retains most of the real information, the rough feature optimization can significantly improve the feature quality and thus greatly improve the detection accuracy.

5.4 Results on PASCAL VOC 2012 and MS COCO

We also verified the performance of GC-FRCN on PASCAL VOC 2012 data set and MS COCO data set. Taking small-scale occluded objects as an example, for VOC 2012 data set, the mAP of Faster RCNN based on VGG-16 network is 64.8%, which reaches 69.4% by combining OSGM and OSIM, increasing by 4.6% than the baseline. Similarly, for the COCO data set, the mAP and mAR of baseline is only 21.7% and 33.1%, which reaches 24.9% and 36.5% by combining OSGM and OSIM.

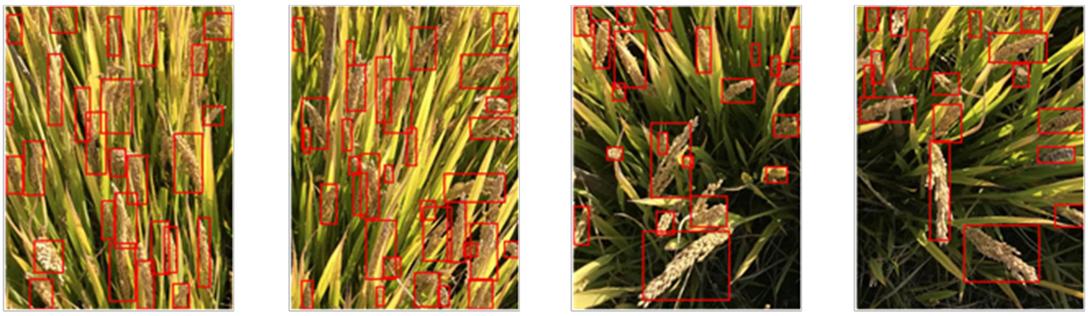
5.5 Results on PANICLE2017

In order to verify the actual detection effect of GC-FRCN on occluded objects, we also applied it to the task of counting rice panicles in the field of current agricultural research. Getting the number of panicles automatic is the key to high throughput rice breeding and intelligent yield measurement, while it is a challenge as the panicle usually locally covered by leaves. The detection effect on rice panicles is shown in Fig. 8a, and LMM (Fernandez et al., 2018), Panicle-SEG (Xiong et al., 2017) and Faster-RCNN are selected as comparison objects. The average counting accuracy and classification accuracy of the four methods are shown in Table 6.

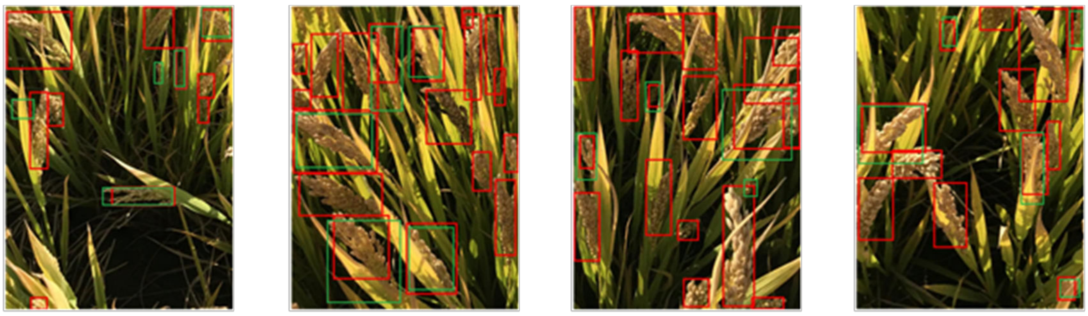
The average counting accuracy and classification accuracy of GC-FRCN are 90.82% and 99.05% respectively, which are 16.12% and 5.15% higher than Faster RCNN algorithm, and about 8% and 4% higher than the similar counting algorithm. As shown in Fig. 8b, we analyze the detection effect of GC-FRCN on blocked rice panicles further in detail. The green box in the visualization results represents the real blocked rice panicles in the image, while the red box represents the detected results by GC-FRCN. The above experimental results firstly verify the hypothesis that occlusion noise will suppress the classification accuracy of object detection. Secondly, it also shows that GC-FRCN can be applied to the detection and counting of rice panicles partially blocked by leaves in complex field scenes by improving the feature quality.

Table 6 Performance comparison of GC-FRCN and other approaches on PANICLE2017 test

Methods	Arch	P_c	P_t
		Average±STD	Average±STD
Faster-RCNN	VGG-16	74.12%±0.19%	93.85%±0.31%
LMM	/	82.16%±0.68%	95.18%±0.36%
Pan-seg	/	82.73%±0.91%	95.45%±0.62%
GC-FRCN（our）	VGG-16	90.82%±0.39%	99.05%±0.20%



(a) Detect effect of GC-FRCN for in-field rice panicle images



(b) Detect effect of GC-FRCN for panicles occluded by leaves locally

Fig8 Detect effect of GC-FRCN for PANICLE2017 test data set

6 Conclusion

In this paper, we propose a detection algorithm for occluded objects based on generative feature optimization, for the problem of low feature quality rising from the external occlusion. Firstly, a quick and low-cost occlusion sample generation module OSGM is introduced, which realized the occlusion simulation and the enhancement of the original training data by screening and discarding the high semantic pixels on the feature map; Secondly, a feature repair module OSIM is introduced, which can repair the occlusion noise as the object's real feature to improve the feature quality. The results of ablation experiments verified the effectiveness of OSGM and OSIM. For the three standard data sets of VOC2007, VOC2012 and COCO, the results show that GC-FRCN can significantly improve the detection accuracy for objects with different scale occlusion. The results of PANICLE2017 data set also show that GC-FRCN can be applied to solve the practical problem of counting rice panicles partially occluded by leaves.

Acknowledge

This paper is supported in part by the National Natural Science Foundation of China (No. 31872847); and the Key Research and Development Plan of Jiangsu Province of China (Modern Agriculture, BE2019383).

Reference

[1]Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[2]He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.

[3]Cai Z, Fan Q, Feris R S, et al. A unified multi-scale deep convolutional neural network for fast object detection[C]//European conference on computer vision. Springer, Cham, 2016: 354-370.

[4]Law H, Deng J. Cornernet: Detecting objects as paired keypoints[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 734-750.

[5]Lu X, Li B, Yue Y, et al. Grid r-cnn[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 7363-7372.

[6]Zhou X, Zhuo J, Krahenbuhl P. Bottom-up object detection by grouping extreme and center points[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 850-859.

[7]Duan K, Bai S, Xie L, et al. Centernet: Keypoint triplets for object detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 6569-6578.

[8]Ouyang W, Zeng X, Wang X. Single-pedestrian detection aided by two-pedestrian detection[J]. IEEE transactions on pattern analysis and machine intelligence, 2014, 37(9): 1875-1889.

[9]Tian Y, Luo P, Wang X, et al. Deep learning strong parts for pedestrian detection[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1904-1912.

[10]Wang X, Xiao T, Jiang Y, et al. Repulsion loss: Detecting pedestrians in a crowd[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 7774-7783.

[11]Zhang S, Wen L, Bian X, et al. Occlusion-aware R-CNN: detecting pedestrians in a crowd[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 637-653.

[12]Bell S, Lawrence Zitnick C, Bala K, et al. Inside-outside net: Detecting objects in context with

skip pooling and recurrent neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2874-2883.

[13]Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

[14]Pepikj B, Stark M, Gehler P, et al. Occlusion patterns for object class detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013: 3286-3293.

[15]Mathias M, Benenson R, Timofte R, et al. Handling occlusions with franken-classifiers[C]//Proceedings of the IEEE International Conference on Computer Vision. 2013: 1505-1512.

[16]Tang S, Andriluka M, Schiele B. Detection and tracking of occluded people[J]. International Journal of Computer Vision, 2014, 110(1): 58-69.

[17]Gidaris S, Komodakis N. Object detection via a multi-region and semantic segmentation-aware cnn model[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1134-1142.

[18]Zhou C, Yuan J. Multi-label learning of part detectors for heavily occluded pedestrian detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 3486-3495.

[19]Noh J, Lee S, Kim B, et al. Improving occlusion and hard negative handling for single-stage pedestrian detectors[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 966-974.

[20]Papageorgiou C, Poggio T. A trainable system for object detection[J]. International journal of computer vision, 2000, 38(1): 15-33.

Viola P, Jones M J. Robust real-time face detection[J]. International journal of computer vision, 2004, 57(2): 137-154.

[21]Felzenszwalb P, McAllester D, Ramanan D. A discriminatively trained, multiscale, deformable part model[C]//2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008: 1-8.

[22]Felzenszwalb P F, Girshick R B, McAllester D, et al. Object detection with discriminatively trained part-based models[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 32(9): 1627-1645.

[23]Dollár P, Appel R, Belongie S, et al. Fast feature pyramids for object detection[J]. IEEE transactions on pattern analysis and machine intelligence, 2014, 36(8): 1532-1545.

[24]Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[25]Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.

[26]He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(9): 1904-1916.

[27]Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

[28]Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[29]Dai J, Li Y, He K, et al. R-fcn: Object detection via region-based fully convolutional networks[C]//Advances in neural information processing systems. 2016: 379-387.

[30]Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[31]Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//European conference on computer vision. Springer, Cham, 2016: 21-37.

[32]Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.

[33]Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[34]Simo-Serra E, Trulls E, Ferraz L, et al. Fracking deep convolutional image descriptors[J]. arXiv preprint arXiv:1412.6537, 2014.

[35]Loshchilov I, Hutter F. Online batch selection for faster training of neural networks[J]. arXiv preprint arXiv:1511.06343, 2015.

[36]Wang X, Gupta A. Unsupervised learning of visual representations using videos[C]//Proceedings of the IEEE International Conference on Computer Vision. 2015: 2794-2802.

[37]Shrivastava A, Gupta A, Girshick R. Training region-based object detectors with online hard example mining[C]//Proceedings of the IEEE conference on computer vision and pattern

recognition. 2016: 761-769.

[38]Wang X, Shrivastava A, Gupta A. A-fast-rcnn: Hard positive generation via adversary for object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 2606-2615.

[39]Xiang P, Wang L, Cheng J, et al. A deep network architecture for image inpainting[C]//2017 3rd IEEE International Conference on Computer and Communications (ICCC). IEEE, 2017: 1851-1856.

[40]Lahiri A, Jain A, Biswas P K, et al. Improving Consistency and Correctness of Sequence Inpainting using Semantically Guided Generative Adversarial Network[J]. arXiv preprint arXiv:1711.06106, 2017.

[41]Yeh R A, Chen C, Yian Lim T, et al. Semantic image inpainting with deep generative models[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 5485-5493.

[42]Dolhansky B, Canton Ferrer C. Eye in-painting with exemplar generative adversarial networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7902-7911.

[43]Pathak D, Krahenbuhl P, Donahue J, et al. Context encoders: Feature learning by inpainting[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2536-2544.

[44]Li Y, Liu S, Yang J, et al. Generative face completion[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3911-3919.

[45]Yu J, Lin Z, Yang J, et al. Generative image inpainting with contextual

attention[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 5505-5514.

[46]Luo W, Li Y, Urtasun R, et al. Understanding the effective receptive field in deep convolutional neural networks[C]//Advances in neural information processing systems. 2016: 4898-4906.

[47]Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[48]Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.

[49]Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, Cham, 2014: 740-755.

[50]Fernandez-Gallego J A, Kefauver S C, Gutiérrez N A, et al. Wheat ear counting in-field conditions: high throughput and low-cost approach using RGB images[J]. Plant methods, 2018, 14(1): 22-34.

[51]Xiong X, Duan L, Liu L, et al. Panicle-SEG: a robust image segmentation method for rice panicles in the field based on deep learning and superpixel optimization[J]. Plant methods, 2017, 13(1): 104-119.