2020 25th International Conference on Pattern Recognition (ICPR)
Milan, Italy, Jan 10-15, 2021

Small Object Detection Leveraging on Simultaneous Super-resolution

Hong Ji*, Zhi Gao*, Xiaodong Liu†, Yongjun Zhang* and Tiancan Mei‡
*School of Remote Sensing and Information Engineering
Wuhan University, Wuhan, China 430079
Email: 2013301220036@whu.edu.cn;gaozhinus@gmail.com;zhangyj@whu.edu.cn
‡Department of Electrical and Computer Engineering
National University of Singapore, Singapore, 117583
Email: xiaodongliu@u.nus.edu
§Electronic and Information School
Wuhan University, Wuhan, China 430072
Email: mtcwlb@aliyun.com

***Abstract*—Despite the impressive advancement achieved in object detection, the detection performance of small object is still far from satisfactory due to the lack of sufficient detailed appearance to distinguish it from similar objects. Inspired by the positive effects of super-resolution for object detection, we propose a framework that can be incorporated with detector networks to improve the performance of small object detection, in which the low-resolution image is super-resolved via generative adversarial network (GAN) in an unsupervised manner. In our method, the super-resolution network and the detection network are trained jointly. In particular, the detection loss is back-propagated into the super-resolution network during training to facilitate detection. Compared with available simultaneous super-resolution and detection methods which heavily rely on low-/high-resolution image pairs, our work breaks through such restriction via applying the CycleGAN strategy, achieving increased generality and applicability, while remaining an elegant structure. Extensive experiments on datasets from both computer vision and remote sensing communities demonstrate that our method obtains competitive performance on a wide range of complex scenarios.**

I. Introduction

Object detection is one of the most important fundamental problems in computer vision, where tremendous efforts have been devoted to. In spite of the great progress achieved in object detection, especially the stunning success of deep convolutional neural network (DCNN) methods proposed in recent years, such as[1–9], the detection performance of small object is still far from satisfactory due to lacking of sufficient detailed appearance to distinguish it from similar objects. As shown in Fig. 1, there is a significant gap between the performance of Faster R-CNN [7] on low-resolution (LR) image and its high-resolution (HR) counterpart. Inspired by [10], it is intuitive to consider SR and detection simultaneously to improve the detection performance. Here, we propose a framework that can be incorporated with object detector networks, resulting in significant improvement.

Prior to the introduction of our work, several key problems should be investigated and clarified. First, whether the peak signal-to-noise ratio (PSNR) that is widely applied in signal processing field for SR evaluation is also the suitable metric for SR evaluation in the detection context. As verified in [12], the answer is no. Given the original LR image, the super-resolved (SR) image with the best PSNR value reported unexpected detection errors. On the other hand, the super-resolved image with the best detection results does not report the best PSNR value. Therefore, our method back-propagates the detection loss into SR network to facilitate detection. Another problem is about the requirement of low-/high-resolution image pairs, which is the base and premise of most available SR methods. Here we downgrade the available images to generate LR counterparts. Benefiting from the strength and

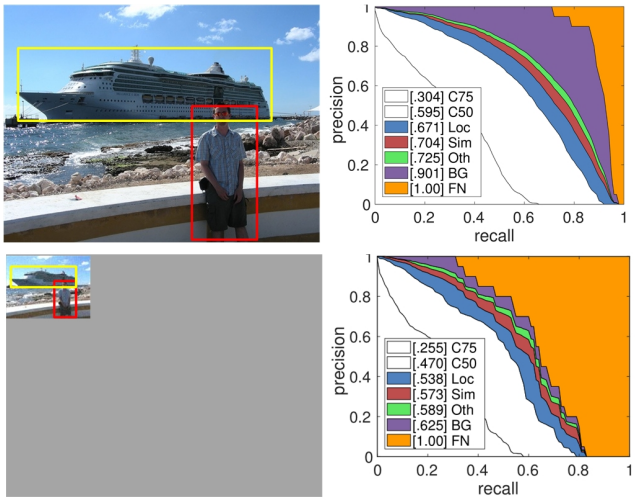

Fig. 1. Example of LR image (bottom-left) and its HR counterpart (up-left)and the overall error analysis of the Faster R-CNN++ detector trained on LR images (bottom-right) and HR images (up-right). We follow [11] to obtain the curves (**C75** means mAP with IoU 0.75. **C50** means mAP with IoU 0.5. **Loc** indicates localization error. **Sim** indicates confusion with similar categories. **Oth** indicates confusion with other categories. **BG** indicates confusion with background. **FN** indicates false negative.) The comparison demonstrates the large gap between the detection performance of HR images and LR images.

[1]This work was partially supported by Peng Cheng Laboratory.

978-1-7281-8808-9/20/$31.00 ©2020 IEEE

2020 25th International Conference on Pattern Recognition (ICPR) | 978-1-7281-8808-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICPR48806.2021.9413058

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

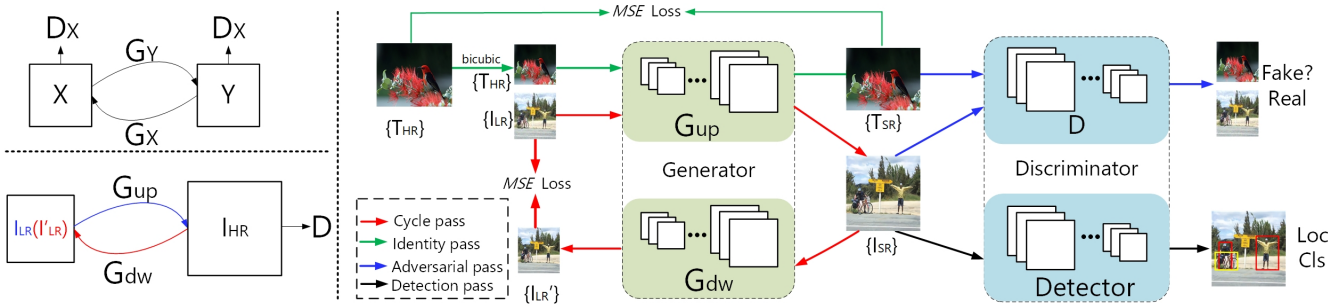

Fig. 2. Illustration of the pipeline of CycleGAN(up left), CycleGAN-like SR network(bottom left) and our framework(right). X and Y represent two image domains. G_X and G_Y are generators. D_X and D_Y are discriminators. I_{LR} is the input LR image, I_{SR} is the super-resolved HR image from I_{LR}, I'_{LR} is of LR generated from I_{SR}. T_{HR} is the HR image provided as reference from other high-quality dataset. T_{LR} is down-sampled version of T_{HR}. T_{SR} is the super-resolved HR image from T_{LR}. Colored arrows represent different parts in the whole framework.

advantages of generative adversarial network (GAN) which can bypass such data preparation and accomplish network training in an unsupervised manner, a few GAN-based SR methods have been reported with promising results and also improved applicability. However, such GAN-based network has not been investigated (or integrated into another network) for detection purpose.

Based on the above discussions which inspire our work from the beginning, we propose a framework for small object detection leveraging on simultaneous super-resolution, in which the CycleGAN-like strategy is investigated to super-resolve the original image to facilitate small object detection and the detection loss is back-propagated into the super-resolution network during training. Our method requires no low-/high-resolution image pairs, achieving increased generality and applicability, while remaining an elegant end-to-end structure. Extensive experiments on datasets from both computer vision (PASCAL VOC 2007, 2012) and remote sensing (DLR Munich) communities demonstrate that our method works effectively on a wide range of complex scenarios.

II. Related Works

The existing related works can be summarized from the following aspects.

Object detection. Encouraged by the success of image classification research [13], the well-known R-CNN work [1] followed the straightforward pipeline of cropping externally obtained regions that potentially include target(s) from the input image and running a deep neural network on such region proposals for final inferring. However, this method is fairly computationally expensive. To alleviate this problem, Fast R-CNN [14] was then proposed, which fed the whole image through a feature extraction network only once so that the (overlapping) crops shared the computational cost for feature extraction. [1] and [14] relied on an independent external operation for proposal generation. Then a Faster R-CNN method [7] which performs detection in two stages has verified that it is possible to apply neural networks to generate region proposals as well. Faster R-CNN is fairly influential and has inspired many successful follow-up works [2–6]. Actually,

TABLE I
Architecture of upsampling Generator G_{up}.

layer	conv	①×16	conv	②	conv	③	conv	③	conv
kernel size	3	3	3	-	3	-	3	-	3
kernel num	64	64	64	-	256	-	256	-	64
stride	1	1	1	-	1	$\frac{1}{2}$	1	$\frac{1}{2}$	1

①,② and ③ represent residual block, element-wise sum, pixelshuffle, respectively.

the region proposal network (RPN) and classifier network can be connected and merged into one, YOLO [9] and SSD [8] are representative works of such one-stage trend.

Image SR. A large number of image SR techniques have been proposed, and readers can refer to survey papers [15, 16] for more information. While the self-similarity based technique is attractive [17, 18] (due to the self-contained nature), most recent techniques utilize external training image pairs for higher performance [19–21]. SR has benefited from recent advances in DCNN. Typically, SRCNN [22] enhanced the spatial resolution of an input low-resolution image by hand-crafted upsampling filters, followed by refinement using DCCN. Based on SRCNN, further improvements were achieved with more advanced network architectures, including residual connections [23] and recursive layers [24]. To further reduce the blurring artifacts, SRGAN [25] has been proposed via combining both perceptual similarity measurement and adversarial losses. To overcome the challenges of preparing low-/high-resolution pairs, a Cycle-in-Cycle GAN structure [26] was proposed to super-resolve the input image in an unsupervised manner.

Simultaneous SR and object detection. Inspired by [10], methods that realize simultaneous SR and object detection have been proposed [12, 27]. In [12], the SSD detection network [8] was fixed, and the detection loss was back-propagated to deep SR network for training. [27] proposed an end-to-end multi-task GAN, in which the generator is an SR network and the discriminator is a multi-task network for real/fake authentication, classification, and localization. However, in both [12] and [27], low-/high-resolution image pairs are required.

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

TABLE II
ARCHITECTURE OF DOWNSAMPLING GENERATOR G_{dw}.

layer	conv	conv×2	residual block ×6	conv×2	conv
kernel size	7	4	3	3	7
kernel num	64	64	64	64	3
stride	1	2	1	1	1

TABLE III
ARCHITECTURE OF DISCRIMINATOR D.

layer	conv	conv	BN	conv	BN	conv	BN	conv
kernel size	4	4	-	4	-	4	-	4
kernel num	64	128	-	256	-	512	-	1
stride	2	2	-	2	-	1	-	1

III. OUR NETWORK

The pipeline of our proposed method is shown in Fig. 2. The processing starts by forwarding the original LR image I_{LR} (of size $w_{LR} \times h_{LR}$) and the details of each component are described in the following.

A. Our CycleGAN-based SR network

In [28], GAN is introduced to generate realistic-looking images from random noise inputs. GAN learns a generator network G and a discriminator D via an adversarial process. The generator G is trained to produce samples to fool the discriminator D, and D is trained to distinguish real from fake images produced by G. Consequently, the objective function of GAN to be optimized is defined as below:

$$\arg\min_{\theta}\max_{\omega}\mathcal{L}_{GAN}(G_\theta, D\omega) \tag{1}$$

here, θ and ω denote the parameters of G and D respectively. Benefiting from the competition strategy, GANs have achieved impressive results in the image generative tasks, such as editing [29], super-resolution [12, 25–27] and style transfer [30, 31]. Specific to the SR problem, the SRGAN loss can be defined as below:

$$\begin{aligned}\mathcal{L}_{SRGAN} = & E_{I_{HR}\sim P_{data}(I_{HR})}[\log(D(I_{HR}))] \\ & + E_{I_{LR}\sim P_{data}(I_{LR})}[\log(1 - D(G_{UP}(I_{LR})))] \\ & + E_{I_{LR}\sim P_{data}(I_{LR})}[||G_{UP}(I_{LR}) - I_{HR}||_2]\end{aligned} \tag{2}$$

where the third term is the pixel-wise mean-squared-error (MSE).

I_{LR}, $G(I_{LR})$ and I_{HR} denote LR image, generated super-resolved image, and real HR image, respectively. However, this constraint requires the availability of sufficient paired low-/high-resolution images for training. We bypass such data preparation via replacing the pixel-wise MSE term with our new term.

Inspired by the CycleGAN[26, 31] strategy, which is shown as up left of Fig. 2, the generator of our GAN is shown in bottom left of Fig. 2, which consists of two sub-generators G_{up} and G_{dw}. We here define the cycle-consistency MSE loss as below:

$$\mathcal{L}_{cyc} = E_{I_{LR}\sim P_{data}(I_{LR})}[||G_{dw}(G_{UP}(I_{LR})) - I_{LR}||_2] \tag{3}$$

where $G_{dw}(G_{UP}(I_{LR}))$ represents the down-sampled image I'_{LR} which has the same resolution as I_{LR}. Based on the cycle-consistency MSE loss term, we try to ensure the similarity between input image and super-resolved image. However, as G_{UP} and G_{dw} are coupled, it is quite difficult to guarantee the convergence of G_{UP} to a network that we want. Thus, leveraging on the additional high-resolution images (which are from some high quality image datasets), we define an identity loss term to ensure the convergence of G_{up}, as below:

$$\mathcal{L}_{Idt} = E_{T_{LR}\sim P_{data}(T_{LR})}[||G_{up}(T_{LR}) - T_{HR}||_2] \tag{4}$$

Here, T_{LR} is obtained via performing bicubic downsampling on the high quality reference image T_{HR}. Here, only G_{UP} is included. Leveraging on both cycle-consistency MSE loss term and identity loss term, our method ensures the similarity between the input image and the super-resolved image without using any prepared low-/high-resolution pairs. The network architectures of our two sub-generators are shown in Table I and II respectively.

Discriminator network D. There are two components in this work. Firstly, we employ the similar network as [26] (described in Table III) to distinguish the real HR images from the generated super-resolved images. For this specific task, MSE loss function is applied in the last convolutional layer. Now, we can formulate our CycleGAN loss, which consists of GAN loss, cycle-consistency MSE loss and identity loss, as below:

$$\mathcal{L}_{cycGAN} = \mathcal{L}_{GAN} + \lambda_1\mathcal{L}_{cyc} + \lambda_2\mathcal{L}_{Idt} \tag{5}$$

Secondly, we employ detector as another discriminator to realize object localization and classification. We study the naive Faster R-CNN using VGG16 as backbone and predicts objects in the last convolutional layer. To demonstrate the effectiveness of our framework, we also make use of advanced techniques for better performance, and train the objective function of multi-task loss as below Eq (6) (here, λ is 1):

$$\begin{aligned}&\mathcal{L}_{Det} = \mathcal{L}_{cls} + \lambda\mathcal{L}_{reg} \\ &\mathcal{L}_{cls} = E_{I_{LR}\sim P_{data}(I_{LR})}[-\log(Det_{cls}(G_{UP}(I_{LR})))] \\ &\mathcal{L}_{reg} = E_{I_{LR}\sim P_{data}(I_{LR})}[smooth_{L1}(Det_{loc}(G_{UP}(I_{LR})), \mathbf{t}_*)], \\ &smooth_{L1}(\boldsymbol{x}) = \begin{cases} 0.5\boldsymbol{x}^2 & \text{if } |\boldsymbol{x}| < 1, \\ |\boldsymbol{x}| - 0.5 & \text{otherwise.} \end{cases},\end{aligned} \tag{6}$$

Here, $\mathbf{t}_*$ is a vector representing the 4 parameterized coordinates of the predicted bounding box.

We now combine the super-resolution and the detection networks to formulate the overall loss function, ad Eq (7):

$$\mathcal{L}_{overall} = \mathcal{L}_{GAN} + \lambda_1\mathcal{L}_{cyc} + \lambda_2\mathcal{L}_{Idt} + \lambda_3\mathcal{L}_{Det} \tag{7}$$

B. Implementation details

The upsampling generator is initialized by the pretrained model released from [32]. The downsampling generator and discriminator are trained from scratch. All the generators and discriminator are trained with Adam optimizer [33]. Their initial learning rates are set to 0.0001 and reduced by a factor of 10 after every 40k iterations. The batch size is 2 and the networks are totally trained for 100k iterations. When

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

training generators, the parameters of discriminator is fixed and objective function is shown as Eq (8), just without the classification loss (3rd term) and localization loss (4th term). In [31], λ is imposed into objectives to control the relative importance of GAN loss and cycle-consistency loss. In this work, we also take λ_1 and λ_2 to control the contribution of GAN loss, cycle-consistency loss and identity loss.

We follow the setting in work [31] to set λ_1 and λ_2 to 10 and 5 respectively. For training discriminator, we fix the generators and the objective function is shown as Eq (9), but without detector loss (2nd and 3rd terms). The detectors are initialized with ResNet-50 and VGG16 trained on ImageNet and trained with SGD optimizer. Initialized learning rate of ResNet50 is 0.0025 and reduced to 0.00025 after 80k iterations, while VGG16 is trained from 0.002 and reduced to 0.0002. They are totally trained for 100k iterations. Here batch_size is 2.

$$\begin{aligned}\arg\min_{G^*}\frac{1}{N}\sum_i ||D(G_{UP}(I^i_{LR}))-1||_2+\\ \frac{1}{N}\sum_i \lambda_1||G_{dw}(G_{UP}(I^i_{LR}))-I^i_{LR}||_2+\\ \frac{1}{N}\sum_i \lambda_2||G_{UP}(T^i_{LR})-T^i_{HR}||_2+\\ \frac{1}{N}\sum_i -\lambda_3\log(Det_{cls}(G_{UP}(I^i_{LR})))+\\ \frac{1}{N}\sum_i \lambda_3[u^i\geq 1](Det_{loc}(G_{UP}(I^i_{LR}),\mathbf{t}^{\mathbf{i}}_*)\end{aligned} \tag{8}$$

$$\begin{aligned}\arg\min_{D^*}\frac{1}{N}\sum_i(||D(G_{UP}(I^i_{LR}))||_2+||D(T^i_{HR})-1||_2)+\\ \frac{1}{N}\sum_i -\omega\log(Det(G_{UP}(I^i_{LR})))+\\ \frac{1}{N}\sum_i \omega[u^i\geq 1](Det_{loc}(G_{UP}(I^i_{LR}),\mathbf{t}^{\mathbf{i}}_*)\end{aligned} \tag{9}$$

After training CycGANSR (called Cyc-SR) network and detecton network, we train them jointly. Its training procedure is as the same as CycGANSR and its objective functions are as Eq (8) and Eq (9) respectively. Here λ_3 and ω are applied to control the importance of detection loss in the whole training process. λ_3, ω are set to 1.

IV. Experiments

We conduct object detection on two representative datasets in computer vision and one additional dataset from the remote sensing community. Moreover, we conduct the comparison against the state-of-the-art detectors, and such detectors synergized with different SR module. Our work and those included for comparison are implemented using Torch and run on an NVIDIA GeForce GTX1080Ti with 12 GB on-board memory.

We first perform experiments on PASCAL VOC [34] that has 20 object categories. We train all the models on VOC 2012 $trainval$ and VOC 2007 $trainval$ respectively, and perform inference on their corresponding test datasets, VOC 2012 $test$ (about $11k$) and VOC 2007 $test$ (about $5k$) respectively. To demonstrate the effect of our work, we down-sample the PASCAL VOC datasets using bicubic kernel to generate LR images. Moreover, we randomly add noise and blur to the images to simulate the practical case. Note that here the low-/high-resolution image pairs have never been used for supervised training, and the original high-resolution images are only used as the ground truth to estimate the upper-limit of different detectors. We evaluate the detection performance using mean Average Precision (mAP), and PSNR is computed to evaluate the SR performance. We focus on the resulting detection accuracy in terms of mAP, and consider the efficiency issue in future. Therefore, we apply the Faster R-CNN networks, both its basic version and its improved version which are termed as Naive Faster R-CNN and Faster R-CNN++ respectively, as our detectors.

A. Results on PASCAL VOC 2007, 2012 datasets

In Table IV, we report the experimental results on the PASCAL VOC 2007 dataset, where the detection results (AP) of each category and the average over all categories (mAP) are reported. In row 2 and 7, Original/FASR++ and Original/FASR demonstrate the result of applying Faster R-CNN++ and Naive Faster R-CNN on the original (ground truth) high-resolution images respectively, which can be applied as the upper-limit to evaluate the performance of different methods. Row 3 to 5 report the results of applying Faster R-CNN++ on the super-resolved images obtained using bicubic, EDSR [32], and CycleGAN respectively. Row 8 to 10 report the results of applying Naive Faster R-CNN on the super-resolved images obtained using different methods. Row 6 and 11 report the results of our method, the same framework but with Faster R-CNN++ and Naive Faster R-CNN respectively. Clearly, the results on the super-resolved images obtained using bicubic interpolation are the poorest. The results using EDSR (the model was released in [32]) are about 5% better than its bicubic counterpart. When the super-resolution is achieved using Cyc-SR, we observe slight improvement over the results using EDSR. Our method outperforms all these methods significantly in each category, and the overall improvement of mAP is more than 3% than the second best method.

In Table V, we report the experimental results on the PASCAL VOC 2012 dataset which is much larger than the previous one, and the data is more challenging. Therefore, the performance of all methods decreases obviously, but the trend of all these methods is nearly as the same as Table IV. In Table V, our method is still the winner on all categories and the improvement of mAP is also about 3% over the second best method. Based on Table IV and V, it is not surprising to find that the results of Faster R-CNN++ are always better than their counterparts using Naive Faster R-CNN, as the detection network has been improved with state-of-the-art techniques.

In Fig. 3, we show the curves of the detection results on VOC 2007 dataset with respect to all, large, medium, and small size objects, respectively. Here we follow the work [35] to define different scales and obtain the curves. For large objects, the mAP of these methods is at the similar level. mAP of our proposed framework even has a slight drop by

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
VOC2007 TEST DETECTION RESULTS OF DIFFERENT METHODS (IOU=0.5).

method	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	house	mbike	person	plant	sheep	sofa	train	tv
Original/FASR++	0.795	0.808	0.810	0.787	0.699	0.664	0.863	0.881	0.887	0.639	0.853	0.748	0.872	0.875	0.800	0.801	0.504	0.792	0.795	0.838	0.771
Bicubic/FASR++	0.595	0.673	0.691	0.522	0.428	0.320	0.692	0.718	0.680	0.410	0.549	0.602	0.646	0.715	0.694	0.692	0.331	0.565	0.615	0.673	0.653
EDSR/FASR++	0.646	0.689	0.738	0.590	0.489	0.401	0.747	0.735	0.742	0.479	0.646	0.653	0.693	0.759	0.702	0.743	0.391	0.659	0.681	0.701	0.657
Cyc-SR/FASR++	0.665	0.727	0.735	0.595	0.492	0.408	0.771	0.756	0.748	0.494	0.693	0.666	0.682	0.769	0.741	0.744	0.385	0.677	0.698	0.724	0.684
Ours/FASR++	**0.699**	**0.755**	**0.762**	**0.644**	**0.543**	**0.474**	**0.784**	**0.775**	**0.800**	**0.506**	**0.740**	**0.689**	**0.769**	**0.819**	**0.806**	**0.767**	**0.434**	**0.708**	**0.723**	**0.774**	**0.700**
Original/FASR	0.759	0.776	0.835	0.747	0.673	0.582	0.844	0.857	0.854	0.597	0.810	0.683	0.824	0.855	0.822	0.788	0.452	0.752	0.734	0.809	0.721
Bicubic/FASR	0.535	0.602	0.647	0.413	0.380	0.251	0.648	0.678	0.617	0.339	0.492	0.598	0.557	0.716	0.669	0.656	0.260	0.491	0.566	0.628	0.569
EDSR/FASR	0.605	0.656	0.710	0.534	0.425	0.325	0.748	0.741	0.668	0.411	0.578	0.637	0.613	0.761	0.714	0.721	0.328	0.553	0.615	0.680	0.610
Cyc-SR/FASR	0.624	0.673	0.717	0.531	0.453	0.362	0.743	0.744	0.709	0.461	0.591	0.628	0.651	0.746	0.737	0.726	0.359	0.605	0.640	0.700	0.648
Ours/FASR	**0.658**	**0.703**	**0.745**	**0.558**	**0.514**	**0.367**	**0.760**	**0.765**	**0.773**	**0.478**	**0.641**	**0.682**	**0.710**	**0.812**	**0.757**	**0.738**	**0.401**	**0.614**	**0.675**	**0.724**	**0.658**

*FASR indicates naive Faster R-CNN detector. *FASR++ indicates Faster R-CNN detector that employs ResNet50 as backbone and investigates FPN architecture for better performance.

TABLE V
VOC2012 TEST DETECTION RESULTS OF DIFFERENT METHODS (IOU=0.5).

method	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	house	mbike	person	plant	sheep	sofa	train	tv
original/FASR++	0.738	0.844	0.803	0.736	0.621	0.601	0.784	0.787	0.912	0.512	0.777	0.602	0.883	0.843	0.830	0.838	0.520	0.758	0.640	0.817	0.649
Bicubic/FASR++	0.544	0.736	0.629	0.478	0.327	0.304	0.657	0.605	0.710	0.333	0.430	0.470	0.673	0.655	0.710	0.698	0.273	0.515	0.466	0.659	0.546
EDSR/FASR++	0.592	0.744	0.659	0.552	0.398	0.400	0.709	0.642	0.761	0.395	0.469	0.506	0.729	0.697	0.708	0.754	0.350	0.576	0.506	0.719	0.563
Cyc-SR/FASR++	0.607	0.773	0.652	0.566	0.409	0.396	0.707	0.662	0.788	0.424	0.509	0.493	0.748	0.707	0.707	0.765	0.379	0.605	0.537	0.726	0.594
Ours/FASR++	**0.637**	**0.792**	**0.711**	**0.589**	**0.436**	**0.432**	**0.735**	**0.683**	**0.815**	**0.437**	**0.576**	**0.539**	**0.773**	**0.731**	**0.778**	**0.786**	**0.392**	**0.640**	**0.552**	**0.756**	**0.597**
original/FASR	0.695	0.818	0.790	0.665	0.552	0.484	0.762	0.738	0.886	0.486	0.730	0.542	0.849	0.808	0.809	0.812	0.451	0.717	0.599	0.780	0.630
Bicubic/FASR	0.485	0.681	0.589	0.400	0.257	0.221	0.637	0.535	0.662	0.270	0.380	0.412	0.596	0.608	0.686	0.643	0.203	0.410	0.416	0.613	0.489
EDSR/FASR	0.550	0.724	0.653	0.480	0.323	0.286	0.682	0.611	0.713	0.334	0.460	0.449	0.693	0.680	0.741	0.720	0.281	0.511	0.454	0.677	0.530
Cyc-SR/FASR	0.566	0.740	0.663	0.504	0.342	0.316	0.689	0.617	0.741	0.360	0.509	0.475	0.691	0.690	0.733	0.719	0.300	0.525	0.471	0.693	0.551
Ours/FASR	**0.594**	**0.759**	**0.684**	**0.534**	**0.398**	**0.320**	**0.708**	**0.628**	**0.768**	**0.383**	**0.549**	**0.518**	**0.729**	**0.711**	**0.770**	**0.737**	**0.342**	**0.557**	**0.530**	**0.701**	**0.553**

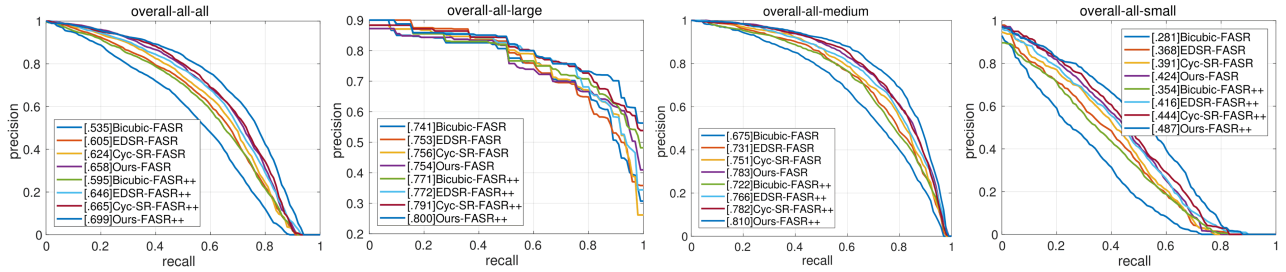

Fig. 3. Overall detection performance on all/large/medium/small objects. The recall and precision are computed with IoU threshold 0.5. We follow work [35] to define the different sizes of objects. From left to right, the figures show mAP over all categories, large, medium and small objects, respectively.

TABLE VI
RESULTS ON LR IMAGES.

Method	mAP	AP_S	AP_M	AP_L
FPN	0.470	0.192	0.626	0.756
SSD	0.412	0.162	0.551	0.613
Ours	**0.699**	**0.487**	**0.810**	**0.800**

AP_S, AP_M and AP_L represents mAP on small, medium, and large objects respectively. Here, the IoU threshold is 0.5.

TABLE VII
AVERAGE PSNR VALUES OF DIFFERENT METHODS.

Method*	Bicubic	EDSR	Cyc-SR	Ours
PSNR	18.49	18.55	25.38	22.42

TABLE VIII
MAP RESULTS ON CHALLENGING IMAGES (IOU=0.5).

Dataset*	original	Bicubic	EDSR	Cyc-SR	Ours
VOC2007/L	0.755	0.520	0.576	0.544	0.590
VOC2012/L	0.711	0.466	0.523	0.488	0.533
VOC2007/E	0.616	0.359	0.298	0.257	0.615
VOC2012/E	0.569	0.304	0.247	0.200	0.564

*L means low-light condition and E represents overexposed condition.

0.02 than that of Cyc-SR when using naive Faster R-CNN network. As to medium objects, gap between the proposed framework and bicubic method climbs by about 10 points. Our framework exceeds Cyc-SR method by approximately 3

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

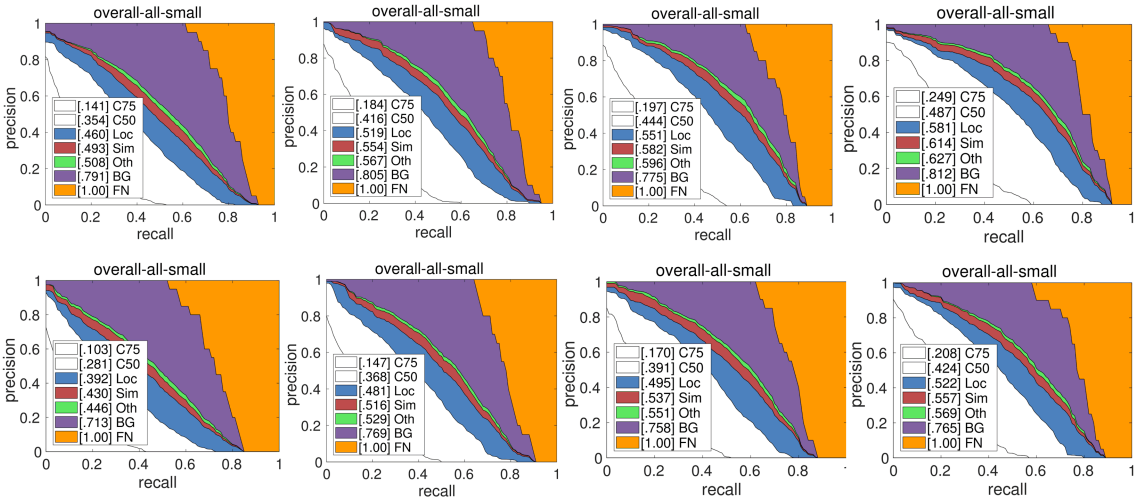

Fig. 4. Overall error analysis of the detection performance on small objects. The meaning of each evaluation setting are the same as Fig. 1. From top left to bottom right, the figure shows results of Bicubic/FASR++, EDSR/FASR++, Cyc-SR/FASR++, ours/FASR++, Bicubic/FASR, EDSR/FASR, Cyc-SR/FASR, ours/FASR, respectively.

Fig. 5. Examples of detection results. Column 1 and column 2 are results that trained and inferred on LR images. Others are trained and inferred on SR images. Green box represents the groundtruth bounding box. Red box represents the detection result.

points. Finally, largest margins are obtained in the small object detection. The proposed framework exceeds bicubic method by dramatically gap 14 points. 7 and 4 points improvements are also obtained compared to EDSR and Cyc-SR methods respectively. In Fig. 4, we show the overall error analysis on small objects detection of different methods. Inaccurate localization, confusion with background and false negative are main causes of detection error. We also compare our framework and the methods that claim to be effective for small object detection, and the results are shown in Table VI. Here we perform the comparison against FPN and SSD. Both the methods take advantage of multi-scale feature extraction and object prediction. For fair comparison, both of them are trained and tested on low-resolution images. It is clear that our method outperforms such methods.

In Table VII, we report the average PSNR values in our previous experiments. Clearly, Cyc-SR outperforms our SR results in terms of PSNR. This result demonstrates that the SR network of our method is detection-driven, which contributes more than other SR components.

In Fig. 5, we display some detection examples in our experiments. Compared to SR methods, SSD and Faster R-CNN++ that are directly trained on LR images, our method is more sensitive to small objects. Imperfect localization occurs in the bicubic and EDSR images due to the noise. We observe that our framework can not only find almost all the objects

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

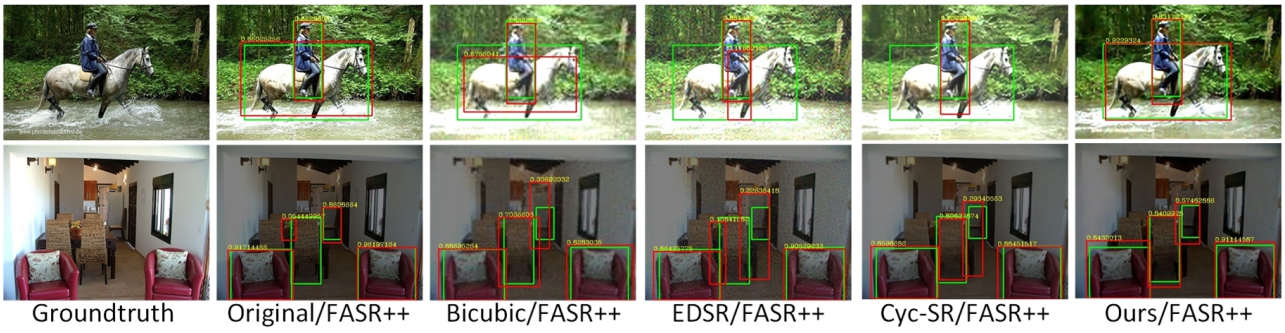

Fig. 6. Examples of the detection results on challenging images. Row 1 and row 2 are overexposed condition. Row 3 and row 4 are low-light condition. Green box represents the groundtruth bounding box. Red box represents the detection result.

TABLE IX
DETECTION RESULTS ON MUNICH DATASET WITH DIFFERENT IOU.

Method*	AP	AP@0.5	AP@0.75
R-FCN	0.321	0.613	0.303
SSD	0.249	0.521	0.212
YOLOv3	0.262	0.574	0.186
FASR++	0.342	0.691	0.292
Bicubic/FASR++	0.487	0.795	0.571
EDSR/FASR++	0.450	0.784	0.530
Cyc-SR/FASR++	0.541	0.801	0.658
Ours/FASR++	**0.599**	**0.889**	**0.684**

but also accurately localize them.

B. Results on challenging images

In order to demonstrate the generalization of our framework, we conduct inference on more challenging scenarios. Here we investigate low-light images and overexposed images (such effects are simulated using software), as shown in Fig. 6. Note that all the models never see those kinds of images during training, and our method achieves the best results.

In Table VIII, we report detection results under challenging conditions. We observe that detector trained on HR images obtained best performance under different settings. Besides, our framework is quite robust for challenging conditions. Especially in overexposed case, our proposed framework outperforms Cyc-SR method by about 35 points.

C. Results on remote sensing images

We further conduct experiments on the remote sensing images where the target (here, vehicle is our target) is small, most less than 30×30 pixels. Here, the DLR Munich dataset [36] is taken at about 1km above the ground over the area of Munich, Germany, using DLR 3K camera system. It contains 20 images (of resolution 5616×3744 pixels), with approximate 13cm ground sampling distance (GSD). The dataset is randomly divided into 410 training and 100 testing images. In Table IX, we report the vehicle detection results of AP at different IoU levels. Obviously, our method achieves the best results for all metrics, obtaining about 5% better AP than the second best result. When the level of IoU is lower, the superiority of our method is more obvious. Some examples of the detection results are shown in Fig. 7.

Fig. 7. Examples of detection results of our method on Munich dataset. Green boxes indicate the correctly detected vehicles, blue and red boxes indicate the missing and false alarms respectively.

V. CONCLUSION

In this work, we propose a framework to facilitate small object detection leveraging on simultaneous super-resolution in an end-to-end manner. Our SR network and detection network are trained jointly. Particularly, the detection loss is back-propagated into the super-resolution network during training to facilitate detection. Compared with the available simultaneous super-resolution and detection methods which heavily rely on low-/high-resolution image pairs, our work breaks through such restriction via applying the CycleGAN strategy, achieving increased generality and applicability. We are going to extend our work to realize instance segmentation of small object, which could provide more valuable information to facilitate precise scene analysis.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[2] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the*

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.

IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3150–3158.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[4] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[5] A. Shrivastava and A. Gupta, "Contextual priming and feedback for faster r-cnn," in *European Conference on Computer Vision*. Springer, 2016, pp. 330–348.

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 1137–1149, 2017.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[9] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[10] J. Shermeyer and A. Van Etten, "The effects of super-resolution on object detection performance in satellite imagery," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[11] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors," in *European conference on computer vision*. Springer, 2012, pp. 340–353.

[12] M. Haris, G. Shakhnarovich, and N. Ukita, "Task-driven super resolution: Object detection in low-resolution images," *arXiv preprint arXiv:1803.11316*, 2018.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[15] K. Nasrollahi and T. B. Moeslund, "Super-resolution: a comprehensive survey," *Machine vision and applications*, vol. 25, no. 6, pp. 1423–1468, 2014.

[16] C.-Y. Yang, C. Ma, and M.-H. Yang, "Single-image super-resolution: A benchmark," in *European Conference on Computer Vision*. Springer, 2014, pp. 372–386.

[17] C.-Y. Yang, J.-B. Huang, and M.-H. Yang, "Exploiting self-similarities for single frame super-resolution," in *Asian conference on computer vision*. Springer, 2010, pp. 497–510.

[18] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5197–5206.

[19] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1865–1873.

[20] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE transactions on image processing*, vol. 21, no. 8, pp. 3467–3478, 2012.

[21] Z. Wang, Y. Yang, Z. Wang, S. Chang, J. Yang, and T. S. Huang, "Learning super-resolution jointly from external and internal examples," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 4359–4371, 2015.

[22] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[23] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654.

[24] ——, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1637–1645.

[25] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[26] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 701–710.

[27] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Sod-mtgan: Small object detection via multi-task generative adversarial network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 206–221.

[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[29] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *European Conference on Computer Vision*. Springer, 2016, pp. 597–613.

[30] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[31] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[32] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[34] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[36] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images." *IEEE Geosci. Remote Sensing Lett.*, vol. 12, no. 9, pp. 1938–1942, 2015.

Authorized licensed use limited to: Wuhan University. Downloaded on March 08,2024 at 07:58:21 UTC from IEEE Xplore. Restrictions apply.