Small-Object Detection with Super Resolution Embedding

Wenyi Tang(✉), Qiucheng Xu, Hui Ding, Lianghao Wu, and Yanyang Shi

State Key Laboratory of Air Traffic Management System and Technology, Nanjing Research Institute of Electronic Engineering, Nanjing 210007, China
tangwenyi@cetc.com.cn

Abstract. The detection performance of ground vehicles on the airport surface in video-sensor based air traffic control surveillance has not been satisfactory compared to larger planes, especially in panoramic images of remote tower applications. Inspired by the success of GAN-based super resolution and oversampling augmentation methods, we apply a new super-resolution network to resize cropped small samples and augment each of those images by resize-and-pasting small objects many times. It allows us to trade off the quantity of the detector on large objects with that on small objects. We propose an architecture with two components: ESRGAN and Detection network. We use residual dense blocks for ESRGAN, and for the detector network, we use one state-of-the-art detector (YOLOv3). Extensive experiments on a public (car overhead with context) dataset and another self-assembled airport surface dataset show superior performance of our method compared to the standalone state-of-the-art object detectors.

Keywords: Object detection · Super resolution · Data augmentation

1 Introduction

Recently, high-density air traffic is becoming a challenge to airport operation. Traditionally, airport surface surveillance methods are based on radar, whose performance suffers from ground clutter, occlusion and multipath effect. In addition, MLAT and ADS-B could only recognize cooperative target. As a result, there is increasingly demand to video-based surveillance at airport surface movement control, which explores in particular the potential of artificial intelligence (AI) and deep learning methods to detect and recognise non-cooperative targets operating on the airfield.

There are several major factors which should be taken into consideration during modeling the detection algorithm. Most of the airfield scenes comprises of a varieties of vehicles with extremely small in size and several relatively larger planes. Moreover, they have variable shapes, multiple scales, orientations, and with complex background which lead to interclass similarities between target and nontarget objects.

Q. Liang et al. (Eds.): Artificial Intelligence in China, LNEE 854, pp. 135–143, 2022.
https://doi.org/10.1007/978-981-16-9423-3_17

To improve the detection of small objects, an approach that draws attention of researchers is to perform super-resolution (SR) to increase the spatial resolution of the images (and thus the size and details of the objects) before performing the detection task. To deal with the lack of details in the low-resolution images, the latest neural network super-resolution (SR) techniques such as CNN-based SR (SR-CNN) [1], Enhanced Deep Residual Super-resolution (EDSR) [2] etc. aim at significantly increasing the resolution of an image much better than the classical and simple bicubic interpolation.

Those aforementioned detectors, which use highly-deep convolution layers, simply ignore the lack of training samples in small object detector training phase. It should be noted that there are relatively fewer images that contain small objects in the dataset, which potentially biases any detection model to focus more on the types of medium and large objects. Secondly, the area covered by small objects is much smaller, implying the lack of diversity of small objects. We conjecture this makes it difficult for the object detection model to generalize to small objects in the test time when their appearance changes.

It is quite straightforward for us to tackle the first issues by copy-pasting those small objects as [3]. The second issue is addressed by high quality resize-and-paste small objects multiple times in each image containing small objects. When pasting each object, we ensure that pasted objects do not overlap with any existing object and increase the resolution of small objects with super resolution methods. This increases the diversity in the locations and scales of small objects while ensuring that those objects appear in correct context.

The contribution of our method can be summarised as:

(1) We propose a super resolution based resize-and-paste algorithm to augment the small objects in the detector training phase.
(2) We propose a novel detector training framework with improved data augmentation, which has better performance in small object detection.

The rest of the paper is organized as follows. In Sect. 2, we first review related work. The details of the proposed method are illustrated in Sect. 3. In Sect. 4, we would presents and discuss the experimental results on two detection benchmark, a public (car overhead with context) dataset and a self-assembled airport surface dataset. Section 5 provides conclusions.

2 Related Work

In this section, we briefly present different deep learning-based methods for super-resolution and detection before focusing on the proposed architecture.

Super Resolution. As mentioned in our Introduction, the two articles [4] give a relatively complete overview of super-resolution techniques based on deep learning. A neural network specialized in super-resolution receives as input a low-resolution image LR and its high-resolution counterpart HR as reference. The network outputs an detail-enhanced higher resolution image $SR = f(LR)$ by minimizing the gap between $f(LR)$ and HR. The simplest architectures are

CNNs consisting of a stack of convolutional layers followed by one or more pixel shift layers. A rearrangement layer allows a change in dimension of a set of layers from the dimension (B, Cr^2, H, W) to (B, C, Hr, Wr), where B, C, H, W represent the number of batches, number of channels, the height and the width of the feature maps respectively and r is the up-sampling factor. Many studies such as [1] already showed a clear improvement of the output image, compared to a classical and simple solution using the bicubic interpolation. An improvement of these networks is to replace the convolutional layers with residual blocks. Part of the input information of a layer is added to the output feature map of that layer. We focus here on the EDSR approach [2] which proposes to exploit a set of residual blocks to replace simple convolutional blocks.

Small Object Detection. Some efforts have been made to tackle the small object detection task by adapting the existing detectors. In [5], Deconvolutional R-CNN was proposed by setting a deconvolutional layer after the last convolutional layer in order to recover more details and better localize the position of small targets. This simple but efficient technique helped to increase the performance of ship and plane detection compared to the original Faster R-CNN. In [6], UAV-YOLO was proposed to adapt the YOLOv3 to detect small objects from unmanned aerial vehicle (UAV) data. Slight modification from YOLOv3 was done by concatenating two residual blocks of the network backbone having the same size.

3 Super Resolution Embedding Augmentation Training

We propose an detection training framework based on super resolution resize-and-paste data augmentation. The proposed framework mainly follow Kisatntal's work [3], which oversample and paste small objects in the training dataset. We would provide an high-level summary of our approach as shown in Fig. 1, leaving the additional details about the more important steps later. Before the training phase, we build the super resolution training pairs from small objects which are cropped in the training frames. Given the proper fine-tuned super resolution model, we resize the small object in different scale and paste the synthetic objects in the original frame.

3.1 Super Resolution Embedding Network

We propose a improved EDSR network, namely ESRGAN. ResNet [7] is proposed by He et al. which suppresses gradient vanishing of deep CNN.

Based on the EDSR network, We replace activation function ReLU [8] with Parametric Rectified Linear Unit [9], which adaptively learns the parameters of the rectifiers and improves accuracy at negligible extra computational cost. Further, we remove residual scaling module [10] from EDSR residual blocks, which plays an important role in making the training procedure numerically stable, as the number of feature maps in EDSR [2] comes up to 256. ESRGAN does not need such trick as it just has 64 filters in each convolution layer (Fig. 2).

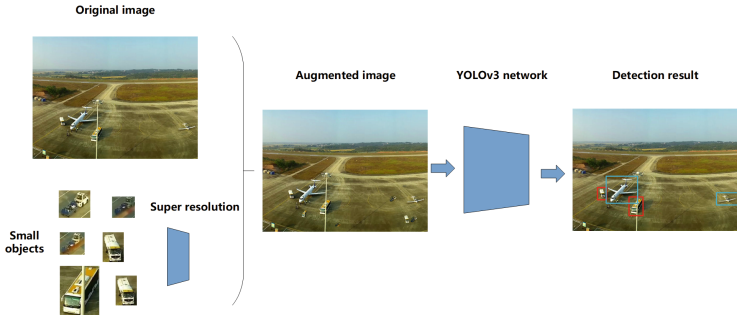

Fig. 1. The flowchart of small object detection based on super resolution embedding data augmentation in airport surface images

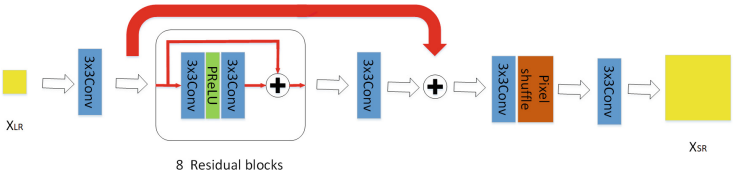

Fig. 2. The network architecture of our prediction network (ESRGAN). It has one 64-filter convolution layer before eight identical residual blocks. Each residual block has two 64-filter convolution layers and a PReLU between them at its residual path. The bending arrow inside residual block is the identity mapping path. One 64-filter convolution layer is appended after residual blocks. An identity mapping the bending arrow) connects the layer before residual blocks and the second layer after eight residual blocks. All the convolution layers above is in 3×3 kernel size. The layer before last convolution layers is combined by one 256 3×3 filters and one pixel shuffle layer. The last layer's filter number agrees to frame channel, which is three for RGB color video.

To make the scaled object consistent with its ground truth under pixel and perceptual criteria, intensity, perceptual constraints are used.

The mean square error (MSE) intensity loss L_{pixel} guarantees the pixel-wise similarity in the pixel RGB space. Specifically, we minimize the $l2$ pixel distance between the resized frame X_{SR} and its ground truth X_{HR} as follows:

$$L_{pixel}(X_{HR}, X_{SR}) = ||X_{HR} - X_{SR}||_2^2 \tag{1}$$

According to Yang [11], the poor perceptual quality of the images obtained by optimizing MSE directly demonstrates a fact: difference in pixel intensity space is not Gaussian additive noise. To tackle this problem, solution is proposed that transferring the image to some feature space where the difference is closer to Gaussian noise. The perceptual loss is designed to optimize network by minimizing the MSE in the feature space produced by VGG-16 [12] as follows:

$$L_{pl} = ||\Psi(X_{SR}; \theta) - \Psi(X_{HR})||^2 \tag{2}$$

where Ψ refers to feature maps at layer conv2_2 in our paper, which is set according to experimental results. Generally speaking, successful supervised networks used for high-level tasks can produce very compact and stable features. In these feature spaces, small pixel-level variation and many other trivial information can be omitted, making these feature maps mainly focusing on human-interested pixels. At the same time, with the deep architectures, the most specific and discriminative information of input are retained in feature space because of the great performance of the network in various tasks. From this perspective, using MSE in perceptual feature space will focus more on the parts which are attractive to human observers with little loss of original contents, so perceptually-pleasing predicted frame can be obtained.

3.2 Adversarial Training

Generative adversarial network was introduced by Goodfellow [13], where images patches are generated from random noise using two networks trained simultaneously. In that work, the authors used a discriminative network D to estimate the probability that a sample comes from the dataset instead of being produced by a generative model G. The two models are iteratively trained so that G learns to generate frames that are difficult to be classified by D, while D learns to tell weather the frames are generated by G. In this work, we follow Ledig's work [14] using a deep CNN binary classifier, which is illustrated in Fig. 3.

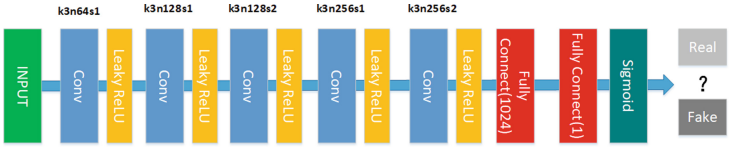

Fig. 3. Architecture of Discriminator Network with corresponding kernel size (k), number of filters (n) and stride (s) indicated for each convolutional layer. The last two Fully-connected layers have 1024 nodes and 1 node respectively.

The training of the pair (G, D) consists of two alternated steps, described below.

Training D: We do one SGD step to D, while keeping the weights of G fixed. Therefore, the loss function we use to train D is

$$L^D_{adv}(X_{HR}, X_{SR}) = L_{bce}(D(X_{HR}), 1) + L_{bce}(D(X_{SR}), 0) \tag{3}$$

where L_{bce} is the binary cross-entropy loss, defined as

$$L_{bce}(x, y) = -\sum_i ylog(x) + (1 - y)log(1 - x) \tag{4}$$

Training G: Keeping the weights of D fixed, we perform one SGD step on G to minimize the adversarial loss:

$$L_{adv}^{G}(X_{SR}) = L_{bce}(D(X_{SR}), 1) \tag{5}$$

3.3 Loss Function

We combine all above constraints regarding appearance, perceptual and adversarial training, into our final loss function of generator:

$$L_G = \lambda_{pixel} L_{pixel}(X_{HR}, X_{SR}) + \lambda_{pl} L_{pl}(X_{HR}, X_{SR}) + \lambda_{adv} L_{adv}^{G}(X_{HR}, X_{SR}) \tag{6}$$

When training D, we use the following loss function:

$$L_D = L_{adv}^{D}(X_{HR}, X_{SR}) \tag{7}$$

3.4 YOLO Detector

After the appropriate ESRGAN is trained according to related objects, we exploit the YOLOv3 object detector as our detection network. It should be noted that the choice of YOLOv3 is optional and could be replaced with other state-of-the-art object detection models. During the standard learning process, the images synthesized by the ESRGAN generator are thus passed to the input of YOLOv3 which calculates the predictions and the loss function (L_{YOLO}) from predicted bounding boxes, whose gradient is back-propagated to update the weights of YOLOv3 network, which seeks to minimize the difference between the detected bounding boxes and the ground truth bounding boxes:

$$L_{YOLO} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_{ij}^{obj} ((x_i - \hat{x_i})^2 + (y_i - \hat{y_i})^2 + (\sqrt{w_i} - \sqrt{\hat{w_i}})^2 + (\sqrt{h_i} - \sqrt{\hat{h_i}})^2) \tag{8}$$

4 Experiments

4.1 Experiment Setup

The proposed ESRGAN is implemented using Tensorflow. All of networks are trained on NVIDIA Titan V GPU with Adam solver. We set minibatch size to 6 and do random horizontal flip and 0,90,180,270 rotation for each training pair. Each pair consists of one clip and its 0.5 resized counterpart. To obtain every loss of a scale that is comparable to others, we set λ_{pixel} to 0.001, λ_{pl} to 0.006, λ_{adv} to 0.05. We firstly do trivial training by setting λ_{adv} to 0 and start adversarial training after 500 iterations warm up. Discriminator net is trained at one tenth of generator's learning rate.

The detector training part is conducted as the standard YOLOv3 routine except the data augmentation which is implemented by resizing 6 random small objects from each training frame and pasting them to different locations without overlapped with existing objects.

4.2 Results on Benchmark Datasets

We evaluate the behavior and the performance of the approaches discussed here using the ISPRS 2D Semantic Labeling Contest dataset [15] and a self-assembled airport surface dataset. The former dataset could be exploited for vehicle detection tasks by retaining only the related components of pixels belonging to the vehicle class. The generated dataset thus contains nearly 10,000 vehicles. The latter dataset is constructed from Jiuhuashan Airport surface surveillance video, which totally is recorded from 2 h of busy flight surface operations. Such video sequences have about 1000 well-labelled objects in three types– planes, trucks and buses (Fig. 4) (Table 1).

Fig. 4. Some frame samples and their ground-truth annotations of the dataset.

Table 1. Quantitative evaluation of ISPRS and airport detection performance using YOLOv3 (confidence threshold of 0.25 and an IoU threshold of 0.25): ESRGAN-2 super-resolution compared to the simple bicubic interpolation and the original size dataset.

Method	ISPRS mAP	Airport mAP
Original	81.82	70.01
Bicubic	71.82	65.01
Proposed	**91.69**	**79.36**

We evaluated this model on an augmented validation set, instead of the original one or the simple bicubic interpolation augmentation one. We saw an increase in the overall object detection performance, suggesting that the trained model effectively "overfit" to small objects. The proposed model yielded better results than the bicubic augmentation, confirming the effectiveness of the proposed strategy of ESRGAN-based resizing small objects.

5 Conclusion

In this paper, we have proposed a robust detection training framework based on the super-resolution resize-and-paste method. To make the detector sensitive to the representation of small objects in a training data, a super resolution resize network is learned from the training dataset and is used to improve the small object diversity in scale and location. We combine the above augmentation method with a state-of-the-art detector YOLOv3 and test our framework on two different challenging benchmarks and experiment results show the relative superiority of proposed algorithm compared to the baseline detector.

Acknowledgments. This work is financially supported by the National Key R&D Program of China, Project Number 2018YFE0208700.

References

1. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 295–307 (2015)
2. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 136–144 (2017)
3. Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., Cho, K.: Augmentation for small object detection. arXiv preprint arXiv:190207296 (2019)
4. Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.H., Liao, Q.: Deep learning for single image super-resolution: a brief review. IEEE Trans. Multimed. **21**(12), 3106–3121 (2019)
5. Zhang, W., Wang, S., Thachan, S., Chen, J., Qian, Y.: Deconv R-CNN for small object detection on remote sensing images. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, pp. 2483–2486. IEEE (2018)
6. Liu, M., Wang, X., Zhou, A., Fu, X., Ma, Y., Piao, C.: UAV-YOLO: small object detection on unmanned aerial vehicle perspective. Sensors **20**(8), 2238 (2020)
7. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 630–645. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_38
8. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML 2010), pp. 807–814 (2010)
9. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)

10. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-ResNet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
11. Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.H.: Deep learning for single image super-resolution: a brief review. arXiv preprint arXiv:180803344 (2018)
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556 (2014)
13. Goodfellow, I., et al.: Generative adversarial nets. Adv. Neural Inf. Process. Syst. **27**, 2672–2680 (2014)
14. Ledig, C., 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, pp. 4681–4690 (2017)
15. Rottensteiner, F., et al.: The ISPRS benchmark on urban object classification and 3D building reconstruction. ISPRS Ann. Photogram. Remote Sens. Spat. Inf. Sci. I-3 **1**(1), 293–298 (2012)