MERGING RECOVERY FEATURE NETWORK TO FASTER RCNN FOR LOW-RESOLUTION IMAGES DETECTION

Ruyi Zhang, Yujiu Yang

Tsinghua University
Shenzhen Key Laboratory of Broad-band Network & Multimedia,
Graduate School at Shenzhen, Tsinghua University, Shenzhen, China

ABSTRACT

The low-resolution (LR) image is one of the obstacles to the improvement of object detection performance. To alleviate the problem, we design a coupled Convolution Neural Networks(CNNs) to recover LR image's information in convolution feature space. One of the CNNs is pre-trained and frozen, which is used for computing label features. Another of the C-NNs is feature recovery network (FRN) which is trainable. Here, FRN is learned by optimizing the distance between features of LR images and its HR images counterpart. In this paper, we merge FRN and Faster RCNN into as a single network by concatenating convolutional features. A large number of experiments show that our merging structure detector has about 2% higher accuracy than the original Faster RCNN on detecting LR images.

Index Terms— Convolutional Neural Network,Object Detection, Low-Resolution, Feature Recovery Network

1. INTRODUCTION

Considering of the systems real-time performance, storage capacity, transmission bandwidth and costs. Low-resolution camera systems exist everywhere in the real world. Based on this, LR images detection has practical significance and application value.Compared with HR images, LR images lose some discriminative details across different objects appearance and locations. It is no surprise that recovering the lost information of LR images in pre-process phase is the first promising solution for a better detector. But in fact,the objective of SR algorithm isn't completely aligned with detection algorithm. As a result, these "two-steps" methods usually have limited performance.

In this paper, we proposed a fusion framework to get better detection performance on LR images. Our method also has two phases. In the first phase, we map HR/LR images into unified convolutional feature space through a pair of C-NNs. HR images are inputted into the pre-trained and frozen label-CNN to get label convolution features. LR images are imported into FRN, which is also pre-trained but is adjustable. In this phase, we learned FRN's parameters to minimize the perceptual loss function[1] based on the convolution feature space distance between HR images and its LR images counterpart. In the second phase, we merged trained FRN and Faster RCNN[2] into a detection network by concatenating convolutional features. It should be pointed out that Faster RCNN in this work is mostly indicated Faster RCNN with shortcut VGG16[3] network: VGG_CNN_M_1024 (VGG_M). And after co-training on the objective dataset, our method can improve original Faster RCNN's detection performance on LR images.

To illustrate our ideas, we summarize the main contributions of the paper as follows: Firstly, we introduced a coupled CNNs (FRN and Label-CNN) to recover LR images information in convolution feature space. Secondly, we designed an effective concatenate structure for merging FRN and Faster RCNN with VGG_M detector. Our experiments show that the proposed network structures can improve original Faster RC-NN's performance on LR images detection.

The rest of the paper is organized as follows. In Sect.2, we describe related works about LR images detection. Sect.3 describes our proposed method. It includes recovering phase and merging phase. Then, we describe our experiments in Sect.4. Sect.5 concludes the paper.

2. RELATED WORKS

Traditional methods are usually based on "two-steps". It includes SR step and detection step. SR algorithms are used for generating an HR image given an LR image. There are many powerful SR algorithms. Such as using compressed sensing [4, 5, 6], convolutional sparse coding[7], et al. Especially, Chao Dong et al. [8] utilized CNN model to solve SR problem.And it got the state-of-the-art result at that time. Then, deeply-recursive convolutional network (DRCN) [9] has better SR result with deeper CNNs. With the Generative Adversarial Network(SRGAN), Christian Ledig et al.[1] bridged the gap to get the finer texture details when super-resolve at large

Thanks the Research Fund for the development of strategic emerging industries by ShenZhen city (No.JCYJ20160301151844537 and No. CXZ-Z20150504163316885)

upscaling factors. After getting super-resolved images, next step is detection. The traditional framework of object detection is using manual features with sliding windows[10, 11]. Recently, CNNs have made impressive improvements in object detection. Region CNN (R-CNN)[12] is one of the earliest CNNs detector. R-CNN need recalculate feature maps for each proposal region, as a result, R-CNN detector is very time-consuming. With the Region Of Interest(ROI) pooling precess, Fast R-CNN [13] increases the detection speed about an order of magnitude. However, the detector speed still depends on the proposal generation. The Faster RCNN[2] has broken this bottleneck with the Region Proposal Network (RPN). YOLO [14] and SSD [15] are belonged to another interesting series detectors, which are single-shot and have high speed.

In fact, existing SR algorithms can't do well in recovering the important discriminative information for detection. In order to recover more useful information and get a more powerful detector. We trained FRN to recover CNNs feature information. Then, we merged FRN to Faster RCNN detector to get better performance on LR images detection.

3. MERGING FRN TO FASTER RCNN DETECTOR

Our model has two phases. The first phase is recovering convolution feature phase. During this phase, we designed coupled CNNs structure(Label-CNN and FRN) and trained FRN for next phase. In VGG case, this phase can be seen from Figure3. The second phase is merging FRN to a detector. Here, we merged FRN to Faster RCNN. The merging structure can be illustrated by Figure1. Then, we co-trained the merging detector on detection datasets. To get more clearly about our model, we describe the details of two phases as follows.

3.1. Phase 1: Recovering convolution feature

Couple CNNs: To recover the LR image's convolution features for detection. We designed a coupled network which is inspired by the perceptual loss in SRGAN[1]. The perceptual loss function can be defined as follows:

$$L_{perc} = \sum_i \|C(G(I_i^{LR})) - C(I_i^{HR})\|_2^2 \qquad (1)$$

C is feature map function. For example, it can be a part of VGG16 or VGG19[3]. In this VGG case, the perceptual loss is defined as VGG loss[1]. With the Euclidean distance between the feature representations, we can get VGG loss math formulation according to the equation 1.

$$L_{VGG_{i,j}} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (C_{i,j}(G(I^{LR})_{x,y}) - C_{i,j}(I^{HR})_{x,y})^2 \qquad (2)$$

With $C_{i,j}$ indicates the feature map which computes through the j-th convolution before the i-th the max pooling layer. Here, $W_{i,j}$ and $H_{i,j}$ means the dimensions of the respective feature maps. The Figure 2 can be more clearly express this loss function.

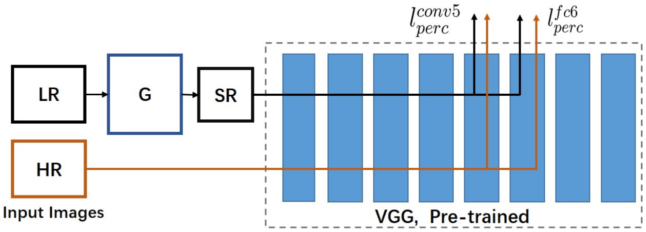

Fig. 2. VGG loss

Because the perceptual loss function has been proved very useful for GAN algorithm[16], and it can ensure the content of generated images consistent with the species category. As mentioned before, the perceptual loss function can be computed in convolution feature space which directly affects CNN's detection performance. Thus, we designed coupled CNNs structure and used the modified perceptual loss function in recovery phase as FRN training objective function. The mathematical formulation is written as follows:

$$L_{perc} = \sum_i \|C_1(I_i^{LR}) - C_2(I_i^{HR})\|_2^2 \qquad (3)$$

C_1 means the convolution features which compute through FRN and C_2 depicts the convolution features which compute through Label-CNN. The convolution features have the same dimensions. If FRN and Label-CNN have the same structure within VGG and use Euclidean distance, the modify perceptual loss function can be rewritten as follows:

$$L_{VGG_{i,j}} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (C1_{i,j}(I^{LR})_{x,y} - C2_{i,j}(I^{HR})_{x,y})^2 \qquad (4)$$

The meaning of symbols can refer to the equation 2. The difference between equations 2 and equations 4 is the former only includes one pre-trained and frozen network, but the laster has two individual CNN netwoks includes pre-trained and frozen Label-CNN and trainable FRN. Figure 3 can more clearly express this coupling structure.

FRN Training: The FRN training objective can be described as follows: Given a training dataset $\{LR^{(i)}, HR^{(i)}\}_{i=1}^{N}$. Our goal is to find the best model $C1$ that accurately regression the feature maps: $C_1(LR^{(i)}) = C_2(HR^{(i)})$. FRN training is a supervised learning process, here, we make HR image feature maps as the label. Then, we trained FRN to recover LR convolution feature. And trained FRN will be used in the next merging phase of detection.

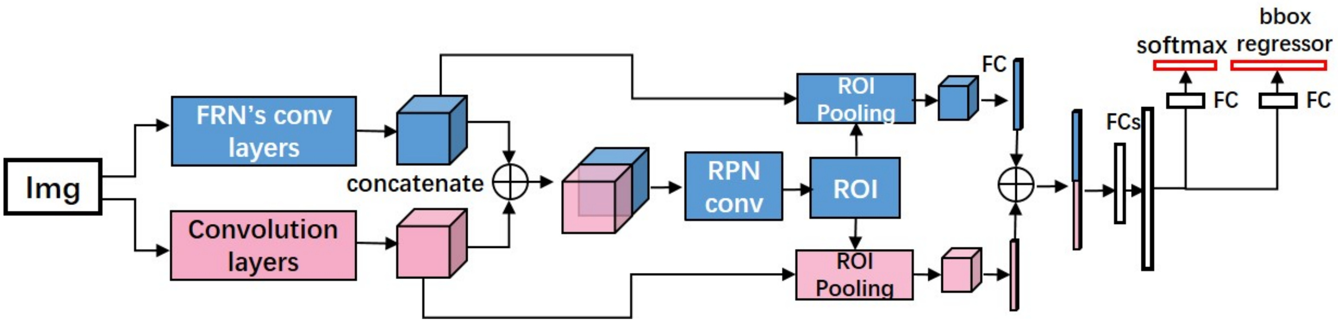

Fig. 1. the structures of merging FRN to faster RCNN

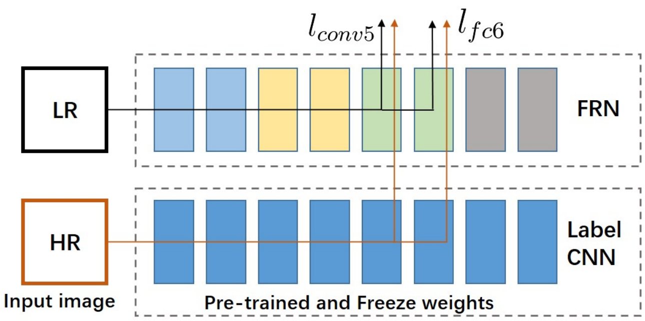

Fig. 3. coupled structure for convolution feature recovery

Although we only need to ensure the last layer dimension of the FRN as same as Label-CNN. To improve convergence, we directly make FRN and Label-CNN using the same structure in this paper. And we find that using the pre-trained weights to initialize all of them can accelerate convergence and improve the performance of the learned FRN. In the other hand, if we train the FRN following the common CNN in training step, it may occur gradient explosive phenomenon. To deal with this phenomena, we utilize the clip gradient [17] to control the weights update in a certain scope.

3.2. Phase 2: Merging FRN to Faster RCNN

Merging Structure: We introduce the details of merging the trained FRN to the Faster RCNN detector. In retrospect, Faster RCNN network has two parts. The first part is RPN and the second part is fast RCNN. Before training, Faster RCNN will load pre-trained weights such as VGG16. Without the pre-training process, the Faster RCNN works poorly. To make merging structure can load mostly pre-trained weights, we not only concatenates feature maps but also concatenate some fully connected (FC) features. The whole merging structure can be illustrated as Figure 1. The cubic means the feature maps. As can be seen from the Figure 1, our structure is concatenated the last convolution feature maps. Then, to load pre-trained FC weights, we put two last convolutional layers to ROI pooling layer individually. At the end concatenate the FC features into the next part of detector network.

Implementation Details We trained all CNNs on a Titan X GPU. We used the end-to-end Faster RCNN training model with VGG_M detector. To better display the effect of our method. Other train parameters set as same as original Faster RCNN. Such as the max iterations equals 90000 for Pascal VOC dataset. We re-scale the images such that their shorter side is $s = 400, 500, 600, 700$ pixels respectively.The base learning rate is 0.001.

4. EXPERIMENT

4.1. Datasets

Detection datasets The performance of the merging FRN detector was evaluated on benchmark datasets Pascal VOC 2007 [18] and Pascal VOC 2012. The PASCAL VOC is the most widely used benchmark dataset for general object detection. In order to get lower resolution datasets, similar to many SR algorithms, we use code to down sample the Pascal VOC images. Then, we used the bi-cubic algorithm to enlarge images to original size to get LR datasets. We evaluate detection results with mean Average Precision (mAP).

FRN datasets Although FRN training is a supervised learning process, we don't need to label dataset manually. There are three steps to prepare the FRN dataset. Firstly, we randomly sample about 240 thousand images from the ImageNet dataset. We treat these images as HR dataset. Secondly, we obtained the LR images by down-sampling the HR images using the bi-cubic kernel with factor r = 4. Thirdly, during the training FRN phase, input HR/LR images will be normalized to 224x224 size. In fact, because original images have different size. Through the three steps above, they don't have the same down-sampling factor.

4.2. The effect of merging FRN detector

In this work, we made the Label-CNN and FRN using the same structure, which is input layer to fc6 layer of VGG_M network. And we set max iterations of FRN is 100000. And clip gradient is set to 35. In the second co-training phase, we

Data	Model	mAP(%)
07× 4LR	F-RCNN	51.5
	F-RCNN+FRN	**53.0**
(07+12)× 4LR	F-RCNN	51.7
	F-RCNN+FRN	**53.7**

Table 1. Detection results on **Pascal VOC 2007 test set** With the down-sampling factor 4(trained on VOC 2007 trainval with the down-sampling factor 4). Here, (07+12) × 4LR means the union set of VOC 2007 trainval and VOC 2012 trainval with the down-sampling factor 4. "Model": original Faster RCNN with VGG_M is abbreviated as "F-RCNN". Merging FRN to Faster RCNN with VGG_M is abbreviated as "F-RCNN+FRN".

Data	Model	mAP(%)
12× 4LR	F-RCNN	46.6
	F-RCNN+FRN	**48.0**
(07+12)× 4LR	F-RCNN	47.1
	F-RCNN+FRN	**48.6**

Table 2. Detection results on **Pascal VOC 2012 test set** with down-sampling factor 4(trained on VOC 2012 trainval with the down-sampling factor 4). The dataset and model can ref to the table1.

concatenated FRN and Faster RCNN's conv5 feature maps and fc6 convolution features. Then we trained and tested the merging structure detector and original detector on the Pascal VOC 2007 LR and Pascal VOC 2012 LR datasets.As can be seen from the Table 1 and Table 2, all performances of merging FRN detector better than original Faster RCNN detector.

We also compared merging structure detector and original detector on the different resolution of Pascal VOC 2007 dataset. The result can be seen from the Table 3. The performance of merging FRN detector better than the original Faster RCNN detector in all experiments. It is easy to find that our method also has improvement on original resolution dataset. The one reason is that we also enlarge images of original resolution dataset during detection training. Another reason is our merging detector has more capacity than the original. In this experiment, the merging structure detector averagely improves 2% mAP than that of the original detector.

4.3. Evaluation of computational efficiency

We improved the model's accuracy at the expense of acceptable speed. Because our model improves the performance with superinducing the FRN. We need to compute more times of convolutional operations.In these experiments, the computation speed of original Faster RCNN is about 25 FPS, and the speed of Faster RCNN with FRN is about 17 FPS.

Data	Model	mAP(%)
07	F-RCNN	61.3
	F-RCNN+FRN	**64.0**
07× 2LR	F-RCNN	58.0
	F-RCNN+FRN	**61.3**
07× 4LR	F-RCNN	51.5
	F-RCNN+FRN	**53.0**
07× 6LR	F-RCNN	46.2
	F-RCNN+FRN	**47.5**

Table 3. Detection results on **different resolution Pascal VOC 2007 test set**. 07 means VOC 2007 original resolution dataset, and the "×2LR, ×4LR, ×6LR" means down-sampling factor is 2,4,and 6, respectively. The model can ref to the table1.

4.4. The effectiveness of trained FRN

We designed this experiment to verify the effectiveness of the trained FRN learning from some useful information. We set two comparison experiments. One of the experiments is merging structure with trained FRN after first feature recovery phase. Another experiment has same merging structure with non-trained FRN which directly used ImageNet pre-trained weights to initiate. As can be seen from the Table 4 , although only the FRN's initial weights are different, the performance of detector with trained FRN is better than that of another detector. The experiment results illuminate that the trained FRN can capture some latent useful information during feature recovery phase and improve the performance of detector.

Data	ImageNet	Trained FRN
$07 \times 4LR$	52.4	**53.0**
$(07+12) \times 4LR$	53.2	**53.7**

Table 4. Detection results on down-sampling **Pascal VOC 2007 test set** with factor 4. The detector is merging FRN to Faster RCNN with VGG_M. "ImageNet": using VGG_M ImageNet pre-trained weights to initialize FRN. "Trained FRN": using FRN which trained through the feature recovery phase.

5. CONCLUSION

We propose a novel framework to get better detection performance on LR images detection. Firstly, we introduced FRN and Label-CNN to recover LR images discriminative information in convolution feature space. Then, we designed an effective concatenate structure for merging FRN to Faster R-CNN detector with VGG_M. A lot of experiments verify that the merging FRN structure can improve the performance of original Faster RCNN for object detection on LR images.

6. REFERENCES

[1] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi, "Photo-realistic single image super-resolution using a generative adversarial network," *CoRR*, vol. abs/1609.04802, 2016.

[2] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017.

[3] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[4] Jianchao Yang, John Wright, Thomas S. Huang, and Yi Ma, "Image super-resolution as sparse representation of raw image patches," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*, 2008.

[5] Weisheng Dong, Lei Zhang, Guangming Shi, and Xiaolin Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. Image Processing*, vol. 20, no. 7, pp. 1838–1857, 2011.

[6] Roman Zeyde, Michael Elad, and Matan Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces - 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers*, 2010, pp. 711–730.

[7] Shuhang Gu, Wangmeng Zuo, Qi Xie, Deyu Meng, Xiangchu Feng, and Lei Zhang, "Convolutional sparse coding for image super-resolution," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 2015, pp. 1823–1831.

[8] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, 2016.

[9] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 1637–1645.

[10] Paul A. Viola and Michael J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[11] Lubomir D. Bourdev and Jonathan Brandt, "Robust object detection via soft cascade," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, 2005, pp. 236–243.

[12] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 580–587.

[13] Ross B. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 2015, pp. 1440–1448.

[14] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 779–788.

[15] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg, "SSD: single shot multibox detector," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, 2016, pp. 21–37.

[16] Alexey Dosovitskiy and Thomas Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 2016, pp. 658–666.

[17] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, "On the difficulty of training recurrent neural networks," in *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, 2013, pp. 1310–1318.

[18] M. Everingham, L. Van Gool, C. K. I. 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, June 2010.