Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.035415

Article

Accelerate Single Image Super-Resolution Using Object Detection Process

Xiaolin Xing[1], Shujie Yang[1,*] and Bohan Li[2]

[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China

[2]Department of Applied Math and Statistics, The Johns Hopkins University, Baltimore, 21211, USA

*Corresponding Author: Shujie Yang. Email: sjyang@bupt.edu.cn

Received: 20 August 2022; Accepted: 14 December 2022; Published: 30 August 2023

Abstract: Image Super-Resolution (SR) research has achieved great success with powerful neural networks. The deeper networks with more parameters improve the restoration quality but add the computation complexity, which means more inference time would be cost, hindering image SR from practical usage. Noting the spatial distribution of the objects or things in images, a two-stage local objects SR system is proposed, which consists of two modules, the object detection module and the SR module. Firstly, You Only Look Once (YOLO), which is efficient in generic object detection tasks, is selected to detect the input images for obtaining objects of interest, then put them into the SR module and output corresponding High-Resolution (HR) sub-images. The computational power consumption of image SR is optimized by reducing the resolution of input images. In addition, we establish a dataset, TrafficSign500, for our experiment. Finally, the performance of the proposed system is evaluated under several State-Of-The-Art (SOTA) YOLOv5 and SISR models. Results show that our system can achieve a tremendous computation improvement in image SR.

Keywords: Object detection; super-resolution; computation complexity; YOLOv5; inference time; objects of interest

1 Introduction and Motivation

Single Image Super-Resolution (SISR) aims to recover an HR image given a Low-Resolution (LR) image, a low-level task in computer vision. The advancement and development of SISR have profoundly influenced people's lives, and it has been directly applied in many fields, including 4K-video [1], face recognition [2], surveillance [3], and medical diagnosis [4]. Since Dong et al. introduced deep Convolutional Neural Networks (CNN) to solve the SISR task in SRCNN [5], numerous researchers have devoted themselves to this field and promoted its development.

Many SR models [6–10] have achieved remarkable performance on restoration quality but pay little attention to computational efficiency or power consumption. Recently, an increasing number of researchers have been working on Video Super-Resolution (VSR), which has high requirements for processing speed. A Mobile AI2021 Challenge report [11] published in CVPR 2021 workshop summarizes the related work of real-time SR on smartphones. How to achieve VSR or real-time SR and apply the technology to low-energy hardware devices such as laptops, smartphones, and embedded devices is a task worthy of exploration. At present, this task faces the following challenges:

- The computing power of the hardware platforms or devices is insufficient or limited, and model migration may encounter difficulties in compatibility.
- Lack of more effective deep learning structures.
- The deeper the model is, the better the performance is. It is difficult to strike a balance between restoration equality and computational efficiency.

First, we are not usually the ones that handle improving the computing power of hardware devices. Second, the early proposed networks [5,6,10] use Bicubic for interpolation preprocessing on LR images, which is computationally expensive and seriously limits processing speed. Subsequently, Dong et al. introduced the deconvolution upsampling module to solve this problem, significantly improving computational performance in FSRCNN [12]. At the same time, Shi et al. [7] proposed the sub-pixel upsampling layer, which proved to be more effective than deconvolution. Kim et al. [6] used 3×3 instead of 5×5 or 9×9 convolution kernels for forward propagation and feature extraction. Third, the inference speed is affected by the model's inherent structure. Deeper networks usually mean more parameters, positively correlated with restoration quality and negatively correlated with inference speed. In the VSR task, the spatio-temporal between adjacent frames is mainly studied [13], and lightweight networks with fewer parameters are designed for acceleration.

SISR is a pixel-dense task. As Fig. 1 shows, SISR is much more computationally expensive than object detection. For a given image, people tend to focus on local objects such as traffic signs, faces, pedestrians [14], or license plates [15]. Mostly, the whole image does not need SR; some background or extraneous things can be ignored or discarded. Based on this hypothesis, we introduce object detection to capture sub-images of interest. YOLOv5, an advanced object detection algorithm based on PyTorch [16], can efficiently get the localization and category of the objects using rectangular bounding boxes. To pursue faster image SR in a limited computation-powering device, we propose a local objects SR system, which uses YOLOv5 to detect objects of interest and then does SR on the gotten sub-images. This way, the objects of interest are captured for the given image, reducing the size of the input image. The computation complexity of SISR is proportional to the image resolution. Compared with direct SR processing, it is more efficient to use an object detection algorithm to obtain local objects with minimal time and do SR on sub-images. An overview of the proposed work is shown in Fig. 2. Preprocessing images with object detection can accelerate the SR process from a new perspective, providing a reference solution for promoting VSR and real-time SR in practical promotion.

We dub our proposed system LO-SR: Local Objects Super-Resolution. Our system has three advantages: First, YOLOv5 is used for object detection, which is lightweight, accurate, fast, and superior to manual operations. Second, our system is end-to-end. Input an image into it and directly output the corresponding HR sub-images of interest. Third, our system is flexible and easy to restructure. Object detection and SR modules can be decided based on the user's needs. At last, we establish a dataset for our experiment to support our work.

Overall, the contributions of our work are mainly in three aspects:

- We propose a local objects SR system named LO-SR, which can reduce the computation complexity of SISR by focusing on objects of interest for a given image.
- We selected multiple object detection and SISR algorithms and did good experiments, which can provide references for practical applications.

- We establish a dataset TrafficSigns500 consisting of 500 traffic images with labeled data.

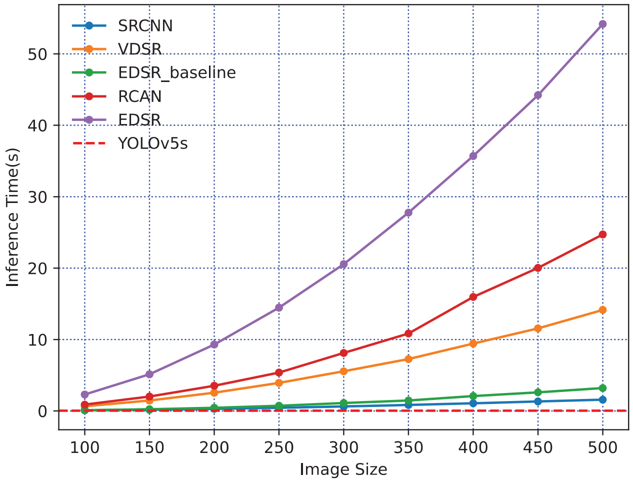

Figure 1: Inference speed comparison between several × 4 SOTA SISRs and YOLOv5s. The inference time of SISRs is roughly linear with the square of the image size (Width × Height), which proves that the SISR task is pixel-dense. On the contrary, the inference time of YOLOv5 is constant and less. YOLOv5s is much more efficient than these SISRs. All works are done on the Intel i5-12500 h central processing unit (CPU) without using the graphics processing unit

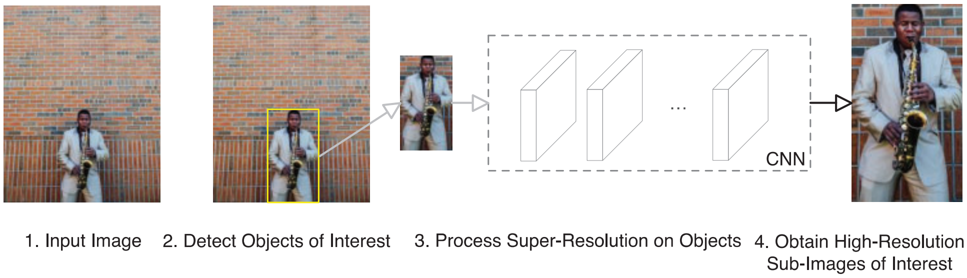
...
CNN
1. Input Image
2. Detect Objects of Interest
3. Process Super-Resolution on Objects
4. Obtain High-Resolution Sub-Images of Interest

Figure 2: Local objects super-resolution overview. Our method (1) Input an image, (2) Capture objects of interest using an object detection model such as YOLOv5s, (3) Process super-resolution only on the objects obtained from the last step, (4) Obtain high-resolution sub-images of interest. Super-resolution is computationally expensive compared to object detection. It can be accelerated by object detection preprocess because the inference speed of super-resolution is proportional to the resolution of the input image

The rest of the paper is arranged as follows. Section 2 describes the related work, mainly introducing SISR and object detection algorithms based on deep learning. In Section 3, we give the formulation and details of our model. Then, we designed an experiment to show the superiority of our model in speeding up the inference speed of super-resolution in Section 4. Finally, we summarize our contributions and discuss future work in Section 5.

2 Related Work

2.1 Single Image Super-Resolution

Traditional SISR methods include the interpolation family [17]: Nearest Neighbor, Bilinear, and Bicubic. Bicubic is often used for downsampling to obtain LR ones given HR images. The pixels at each position are interpolated based on the pixels around them. Its results are smooth, and it has stable restoration quality, but the details of the image are lost. Bicubic is often used as a benchmark for SR performance comparison.

Another kind of method is learning-based, and it tries to construct the mapping from LR images to HR images. The representative conventional machine learning algorithm is sparse-coding [18,19], which involves multiple steps in its solution pipelines and can be summarized as image cropping, preprocessing, encoding, reconstruction, and aggregation.

CNN has made remarkable achievements in image classification [20] and was later introduced to solve this problem, image SR. Dong et al. proposed a three-layer deep convolutional network for the first time. They gave a theoretical basis by establishing a relationship between the proposed method and the traditional spare-coding SR method.

Deep convolutional network models recover an HR by constructing a deep CNN network to perform feature extraction, nonlinear mapping, upsampling, and image reconstruction from a given LR. Thanks to backpropagation, all modules can be uniformly learned and optimized. This method achieves excellent image reconstruction quality with hardware technology development and massive data.

Kim et al. [6] first introduced residuals into the image SR task, and then ResNet [21] variants emerged in an endless stream. EDSR [8] removed the Batch Normalization (BN) layer of ResNet in classification, expanded the model size, and championed the NTIRE2017 challenge on image SR. Then, the models represented by RCAN [9] and HAN [22] introduced the attention mechanism into this task, and the restoration quality was slightly improved. With the introduction of ViT [23], Transformer is applied to computer vision. Compared with the existing SOTA models, SwinIR [24] achieved better performance while reducing the number of parameters by 67%. At the same time, some branches of SR tasks have also been advanced, such as lightweight models [25], arbitrary upscale [26], asymmetric upscale [27], and generative adversarial networks [28].

2.2 Object Detection

Object detection has recently been widely studied and explored as one of the most fundamental and challenging computer vision tasks. Given an image, generic object detection aims to localize existing objects with rectangular bounding boxes and classify them with confidence values. It can be applied in some specific application fields, including face recognition, pedestrian detection, product recognition [29], vehicle detection and tracking [30], etc. In addition, object detection is an essential part of many other computer vision tasks such as instance segmentation. Multi-scale object detection [31–33] is currently a hot topic in object detection tasks. It has higher requirements for model design and needs to consider more complex scenarios.

In the past twenty years, the development of object detection has roughly experienced two historical periods: the traditional object detection period (before 2014) and the object detection period based on deep learning (after 2014). Benefiting from the boom of computing power and the significant breakthrough of deep learning, the latter has become the leading for object detection.

Here we only introduce object detection algorithms based on deep learning, which can be divided into two categories: region proposals based and regression or classification based. The methods of region proposals based are two-stage, which can divide the object detection task into a combination of two subtasks, region proposals and classification. The representative algorithms include R-CNN [34], SPP-net [35], Fast R-CNN [36], and Faster R-CNN [37], etc. Firstly, a specific module generates or decides proposal regions, and another module processes the classification task on the former outputs. This two-stage method is characterized by high accuracy but slow speed. The methods of regression or classification based are one-stage, pursuing high speed with only one module directly predicting the categories and localizations. In practical usage, the methods of regression-based are more popular. YOLO [38] is the most widely used among all object detection algorithms. According to the table given by ULTRALYTICS, the latest YOLOv5 model can reach 156fps on CPU V100 b1. Moreover, through the improvement and optimization of multiple versions, the detection accuracy has also been greatly improved, which can meet the needs of most practical applications. This paper uses the latest YOLOv5 as the object detection module, which contains several optional parameter weights: v5n, v5s, v5m, and v5l. For more details, you can reference https://github.com/ultralytics/yolov5.

3 Methodology

Our proposed LO-SR consists of two modules. The first captures objects of interest. These objects are cropped from the original image and form an SR candidate set. The second module does SISR on the candidate set to generate counterpart HR images. In this section, we give a formulation for this task, then describe the implementation details of the system.

3.1 Formulation

Consider an image, which can be represented as $\mathbf{I}$. The image's resolution is usually a little large, while the resolution of objects in $\mathbf{I}$ can be said to be small (mostly 30×30). There are n proposal objects in the image. First, object detection is performed on $\mathbf{I}$, and obtain multiple LR sub-images, which are objects of interest represented as $\mathbf{I}^{LR} = \{l_1, l_2, \ldots, l_n\}$. Then crop these sub-images from the original image. Next, cropped sub-images are fed into the SR module, and we can get $\mathbf{I}^{HR} = \{h_1, h_2, \ldots, h_n\}$, where l_i corresponds to h_i, $1 \leq i \leq n$. F_{od} and F_{sr} are respectively used to represent the object detection process and image SR process, without considering other input parameters, which can be expressed as simply:

$$\mathbf{I}^{\mathrm{LR}} = F_{od}\left(\mathbf{I}\right) = \{l_1, l_2, \ldots, l_n\} \tag{1}$$

$$\begin{aligned}\mathbf{I}^{HR} &= F_{sr}\left(\mathbf{I}^{LR}\right)\\ &= F_{sr}\left(\{l_1, l_2, \ldots, l_n\}\right)\\ &= \{h_1, h_2, \ldots, h_n\}\end{aligned} \tag{2}$$

Only considering the input and the output of our system, the above equations can be combined as:

$$\begin{aligned}\mathbf{I}^{HR} &= F_{sr}\left(F_{od}\left(\mathbf{I}\right)\right)\\ &= F_{losr}\left(\mathbf{I}\right)\end{aligned} \tag{3}$$

where F_{losr} defines our end-to-end local objects SR operation. Therefore, the selection of the object detection module and SR module directly determines the system's performance, mainly reflected in the accuracy of acquiring objects of interest, image restoration equality, and inference speed.

3.2 LO-SR Details

Both the objection detection module and SISR module are pluggable. For a tremendous LO-SR system, the computational cost of object detection is much less than that of SISR. As shown in Fig. 1, object detection is very efficient and is not the main factor affecting inference speed. We compared multiple object detection algorithms and used YOLOv5 as the lightweight object detection module. It has excellent mean Average Precision (mAP) value, is based on PyTorch implementation, and is easy to deploy on various devices. YOLOv5 provides five modules for us to select with different parameters from 1.9 M to 86.7 M.

So far, there have been a variety of SOTA algorithms. As our module, we chose several SISR algorithms published in top conferences with high quotations (such as CVPR, ICCV, and ECCV). Simply put, our contribution is applying the published algorithms and providing a scheme and idea for all practical applications.

Our system is a fully connected topology, as shown in Fig. 3, where appropriate modules are enabled for object detection accuracy, image restoration quality, and inference speed. A system workflow is given in Fig. 4 and consists of the testing environment and production environment. In testing, users establish datasets as required for offline training. Alternative sets can be obtained based on specific demand, in which all LO-SR systems meet the user demand. Finally, one of the alternatives is decided and deployed to the hardware device to run the real-time online work.

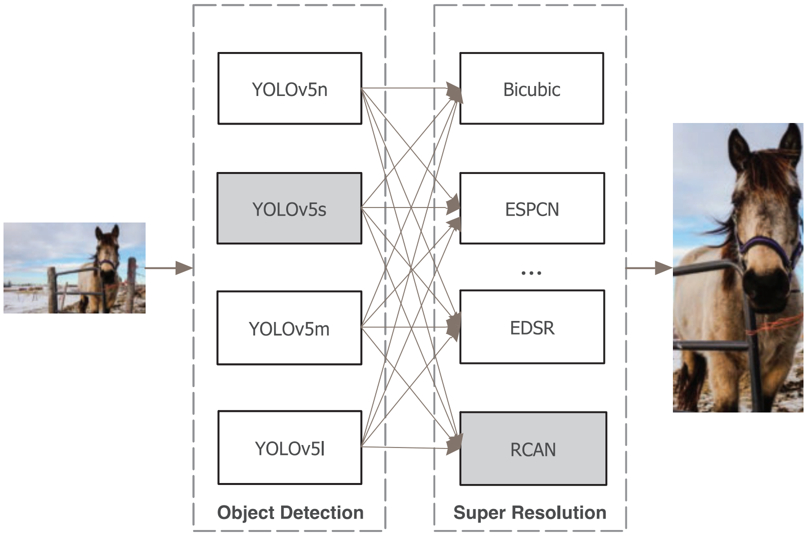

Figure 3: A review of our LO-SR system. Object detection modules and super resolution models are pluggable, and YOLOv5s and RCAN are enabled in this work. You can enable two modules according to your needs and only deploy them in hardware devices

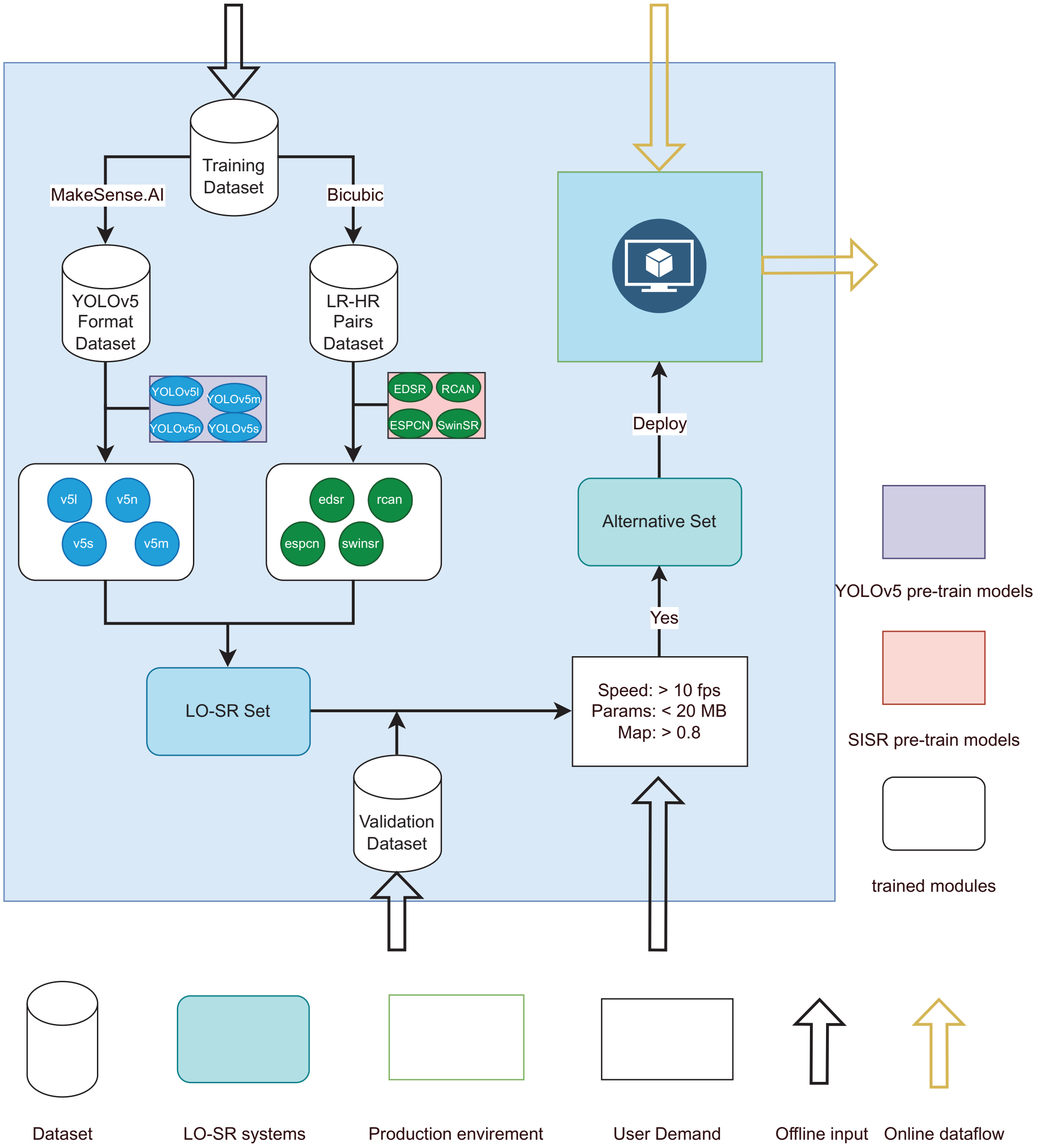

Figure 4: The workflow of our LO-SR system generally consists of offline training and online work

4 Experiment

In this section, a traffic sign SR task is designed to present the superiority of our proposed method. Firstly, we briefly describe the training in object detection and SISR methods. Second, some evaluations show the value of our system.

4.1 Datasets and Implementation Details

The training is divided into two steps: object detection training and image SR training. The former movement uses the dataset according to specific application scenarios to obtain the objects of interest. The latter is trained on many images and often has a high generalization ability.

Traffic Signs 500 is a dataset proposed by us in which all images are from work done by Zhang et al. [39]. We used MakeSense.AI [40] to create the dataset in YOLOv5 format. Traffic Signs 500 consists of 400 training images and 100 validation images. Among them, each image contains several traffic signs. The typical value for the image resolution in this dataset is 1024 × 768. In this dataset, all images are taken from inside the car. These traffic signs are a small percentage of the overall image and are low resolution, so the drivers sometimes can't see them clearly and accurately. At the same time, the drivers are not interested in other objects such as billboards, telegraph poles, and trees presented on the car monitor. Therefore, this dataset is suitable for showing the value of LO-SR, in which traffic signs are divided into three categories: Caution, Prohibitory, and Guide. We trained several YOLOv5 pre-train models on this dataset to support the object detection module. Fig. 5 shows some results of YOLOv5s on samples from Traffic Signs 500.

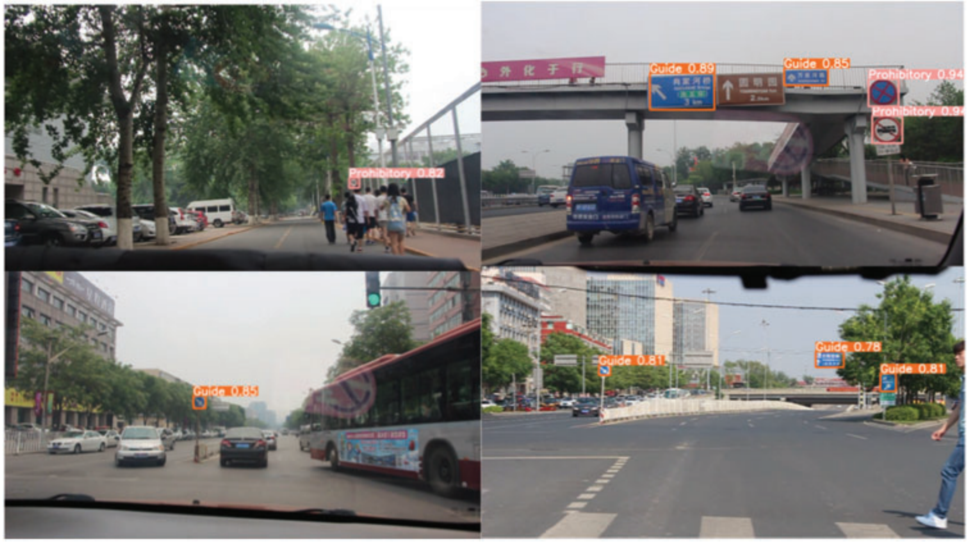

Figure 5: Object detection results of some samples from Traffic Signs 500

We selected several SISR algorithms as the image SR module of LO-SR, including SRCNN [5], VDSR [6], MemNet [10], EDSR [8], and RCAN [9]. The DIV2K [41] dataset is a high-quality (2K resolution) image dataset for the image SR task. This dataset consists of 800 training images, 100 testing images, and 100 validation images. We introduced pre-train models with warm-start, trained two upscale, × 2 and × 3 on the training set for 100 epochs, and then evaluated on the validation set, achieving close results to those in related papers.

4.2 Evaluation

Our proposed system aims to obtain objects of interest and accelerate image SR using the object detection process. Multiple SISR methods are introduced to compare the inference speed of the conventional image SR and our LO-SR. In general, the evaluation metrics for image SR are Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM), and the evaluation metric for object detection is mAP. Our proposed system is neither an image SR model nor an object detection model, so there is no need to compare these metrics. The inference time is all we care about.

The evaluation compares the inference speed using LO-SR and SISRs on the same validation dataset in TrafficSigns500. Table 1 shows multiple comparisons. In SRCNN [5], VDSR [6], and MemNet [10] models, the input LR images are pre-processed by Bicubic interpolation and have a consistent resolution with the ground truth HR images.

Table 1: The inference speed(s) of multiple LO-SR systems on Intel i5-12500 h CPU 4.5 GHz

YOLO/SR	mAP@0.5	params(M)	Scale	Bicubic	SRCNN [5]	VDSR [6]	MemNet [10]	RCAN [9]	EDSR [8]
-/-	-/-	-/-	× 2	5.369	14.34	19.94	50.98	90.55	192.7
			× 3	20.10	25.34	33.21	108.4	100.3	198.2
v5n	0.953	1.9	× 2	0.2301	0.3423	0.6184	2.698	3.870	7.818
			× 3	0.3172	0.6101	1.148	4.541	4.291	8.261
v5s	0.985	7.2	× 2	0.1962	0.2240	0.5051	1.977	3.885	7.826
			× 3	0.2172	0.3231	0.9212	4.204	4.178	8.143
v5m	0.979	21.2	× 2	0.2502	0.3032	0.5981	2.309	4.116	7.911
			× 3	0.5254	0.6324	1.420	5.097	4.239	8.377
v5l	0.973	46.5	× 2	0.3051	0.3602	0.6723	2.896	4.217	8.829
			× 3	0.5850	0.6941	1.350	5.247	4.731	9.219

All numerical results are computed on the personal computer with Intel i5-12500 h CPU 4.5 GHz and RAM 16 GB 4800 MHz using the same timer. As you can see from the first row of the table, it is computationally expensive to do SR on an image with a high resolution directly. For RCAN [9], the inference time of × 2 scale is up to 90 s. In our LO-SR model, traffic signs are obtained using the YOLOv5 module. The results on the validation dataset show that traffic sign detection accuracy is excellent; all mAP@0.5 values are higher than 0.95. The inference speed is greatly improved using our LO-SR. Overall, the LO-SR improves the image SR speed by at least 20 times. We also tested on Intel i7-12800 HX CPU 4.8 GHz and RAM 32 GB 3200 MHz; the results are shown in Table 2. Almost all the inference time is less than 1 s. Our LO-SR system can achieve real-time SR on this dataset with high resolution only using generic computing power devices, which is impossible for conventional image SR algorithms. A visual result of LO-SR is shown in Fig. 6.

Table 2: The inference speed(s) of multiple LO-SR systems on Intel i7-12800 HX CPU 4.8 GHz

YOLO/SR	mAP@0.5	params(M)	Scale	Bicubic	SRCNN [5]	VDSR [6]	MemNet [10]	RCAN [9]	EDSR [8]
-/-	-/-	-/-	× 2	1.213	3.123	4.234	10.12	19.23	40.13
			× 3	3.231	4.328	6.567	23.34	20.73	41.04
v5n	0.953	1.9	× 2	0.0392	0.0512	0.0934	0.5189	0.5314	1.369
			× 3	0.5001	0.1038	0.5231	0.8672	0.7482	1.387

(Continued)

Table 2 (continued)

YOLO/SR	mAP@0.5	params(M)	Scale	Bicubic	SRCNN [5]	VDSR [6]	MemNet [10]	RCAN [9]	EDSR [8]
v5s	0.985	7.2	× 2	0.0410	0.06831	0.9813	0.4238	0.7182	1.521
			× 3	0.04651	0.05320	0.1972	0.8219	0.7906	1.582
v5m	0.979	21.2	× 2	0.05211	0.05672	0.1065	0.4123	0.9834	1.968
			× 3	0.1043	0.1478	0.2864	1.292	1.1904	2.317
v5l	0.973	46.5	× 2	0.04180	0.07531	0.1442	0.6134	0.9671	1.915
			× 3	0.1300	0.1632	0.2211	1.028	1.082	2.128

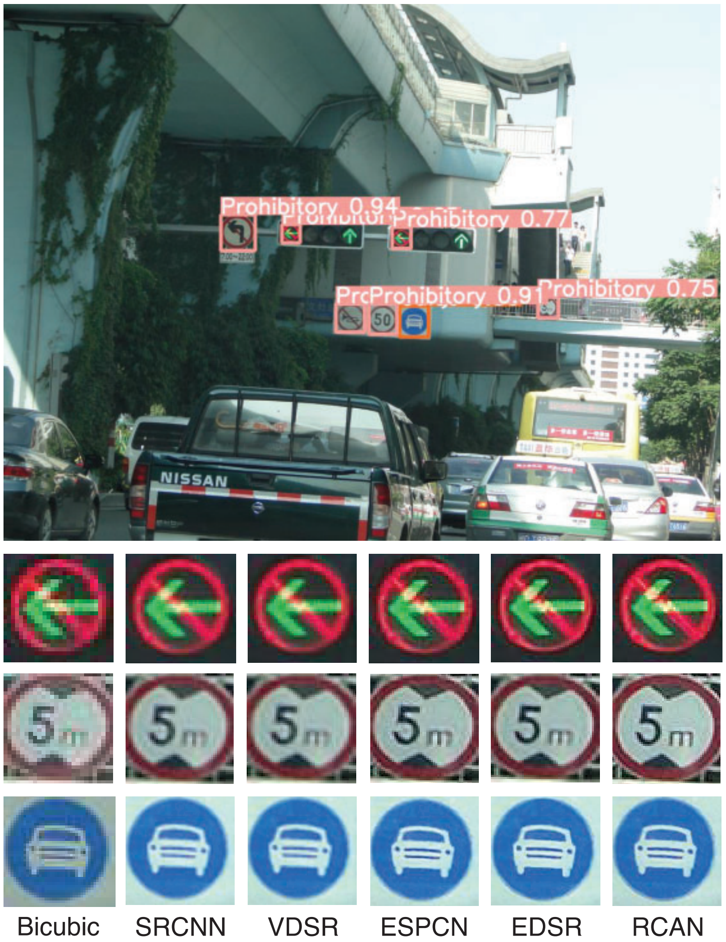

Figure 6: Visual results for LO-SR upscale × 2 in chaotic traffic surroundings where the objects of interest are unclear. Our LO-SR system can accurately capture objects of interest and obtain corresponding HR images. This way, the information conveyed by the visual image is easier to identify

5 Conclusion and Future Work

5.1 Conclusion

In this paper, we argue that SISR is computationally expensive compared to object detection, then propose a two-stage object detection and image SR system to accelerate image SR. By cropping objects of interest, we significantly reduce the size of the input images fed to SR to speed up this process. Our eyes always focus on local objects in one image, and our LO-SR system can help us quickly observe these objects of interest. We establish a dataset, Traffic Signs 500, to support our experimental section. In this dataset, we focus on traffic signs on an image taken by a car camera, which tends to be low resolution. Our proposed system achieved excellent performance in accelerating SR.

5.2 Discussion and Future Work

Our work combines two tasks in the realm of computer vision, object detection, and SISR. Our contribution is mainly to reduce the computational complexity of SISR tasks and provide a new perspective for promoting this technology to practical applications. We do not design a new model or architecture from scratch but use the existing algorithms for secondary innovation. Through experiments, we show the value of our method, which can significantly reduce power consumption, which is extremely important for some special-purpose devices. Overall, our work can provide a research viewpoint for accelerating image SR.

However, our method also has shortcomings and improvements. First, obtaining all objects of interest has a crucial impact on the effectiveness of our system. The images fed to the SR module are the sub-images obtained by the object detection process. It cannot get the expectations if the outputs are missing or inaccurate. One optimization scheme that can be thought of is to use manual bounding to remedy. Second, when many objects are detected in a given image, the number of sub-images to crop may be large. In this case, the effect of our system is insignificant, and our attention would be distracted. Third, the object detection and SR modules are designed or decided separately in our system. We believe collaborative design and optimization of these two modules can improve the system's performance. Fusing both subtasks into one optimization task and designing a unified model is what we would do in future work.

Acknowledgement: Thanks to my co-authors for their assistance and instructive help. My best thanks to my advisor Changqiao Xu. I am very grateful to Zhuoyuan Li for his guidance and support in drawing. Thanks for the editor's work and consideration of the publication of our paper. On behalf of my co-authors, we would like to express our great appreciation to the editor and reviewers.

Funding Statement: This work is supported by the National Natural Science Foundation of China (NSFC) under Grant No. 62001057; by Beijing University of Posts and Telecommunications Basic Research Fund, 2021RC26; by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61871048 and 61872253.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. Cao, C. Wang, C. Song, Y. Tang and H. Li, “Real-time super-resolution system of 4k-video based on deep learning,” in *Proc. of the 2021 IEEE 32nd Int. Conf. on Application-Specific Systems, Architectures and Processors (ASAP)*, NJ, USA, pp. 69–76, 2021.

[2] Y. Kortli, M. Jridi, A. A. Falou and M. Atri, “Face recognition systems: A survey,” *Sensors*, vol. 20, no. 2, pp. 342, 2020.

[3] P. Shamsolmoali, M. Zareapoor, D. K. Jain, V. K. Jain and J. Yang, “Deep convolution network for surveillance records super-resolution,” *Multimedia Tools and Applications*, vol. 78, no. 17, pp. 23815–23829, 2019.

[4] M. Bakator and D. Radosav, “Deep learning and medical diagnosis: A review of literature,” *Multimodal Technologies and Interaction*, vol. 2, no. 3, pp. 47, 2018.

[5] 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.

[6] J. Kim, J. K. Lee and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in *Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 1646–1654, 2016.

[7] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken *et al.,* “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 1874–1883, 2016.

[8] B. Lim, S. Son, H. Kim, S. Nah and K. M. Lee, “Enhanced deep residual networks for single image super-resolution,” in *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Honolulu, HI, USA, pp. 1132–1140, 2017.

[9] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong *et al.,* “Image super-resolution using very deep residual channel attention networks,” in *European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 286–301, 2018.

[10] Y. Tai, J. Yang, X. Liu and C. Xu, “MemNet: A persistent memory network for image restoration,” in *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 4539–4547, 2017.

[11] A. Ignatov, A. Romero, H. Kim and R. Timofte, “Real-time video super-resolution on smartphones with deep learning, mobile AI 2021 challenge: Report,” in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Virtual, pp. 2535–2544, 2021.

[12] C. Dong, C. C. Loy and X. Tang, “Accelerating the super-resolution convolutional neural network,” in *European Conf. on Computer Vision (ECCV)*, Amsterdam, The Netherlands, pp. 391–407, 2016.

[13] M. Chu, Y. Xie, J. Mayer, L. Leal-Taixé and N. Thuerey, “Learning temporal coherence via self-supervision for GAN-based video generation,” *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 1–13, 2020.

[14] M. Sukkar, D. Kumar and J. Sindha, “Real-time pedestrians detection by yolov5,” in *Proc. of 2021 12th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, pp. 1–6, 2021.

[15] Y. Jamtsho, P. Riyamongkol and R. Waranusast, “Real-time license plate detection for non-helmeted motorcyclist using YOLO,” *ICT Express*, vol. 7, no. 1, pp. 104–109, 2021.

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury *et al.,* “Pytorch: An imperative style, high-performance deep learning library,” in *Proc. of Advances in Neural Information Processing Systems (NIPS)*, Vancouver, BC, pp. 8024–8035, 2019.

[17] F. Zhou, W. Yang and Q. Liao, “Interpolation-based image super-resolution using multisurface fitting,” *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3312–3318, 2012.

[18] M. Alain and A. Smolic, “Light field super-resolution via LFBM5D sparse coding,” in *Proc. of 2018 25th IEEE Int. Conf. on Image Processing (ICIP)*, Athens, Greece, pp. 2501–2505, 2018.

[19] J. He, L. Yu, Z. Liu and W. Yang, “Image super-resolution by learning weighted convolutional sparse coding,” *Signal Image and Video Processing*, vol. 15, no. 5, pp. 967–975, 2021.

[20] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang *et al.,* “Review of image classification algorithms based on convolutional neural networks,” *Remote Sensing*, vol. 13, no. 22, pp. 4712, 2021.

[21] K. He, X. Zhang, S. Ren and J. Sun, “Deep residual learning for image recognition,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 770–778, 2016.

[22] B. Niu, W. Wen, W. Ren, X. Zhang, L. Yang *et al.,* “Single image super-resolution via a holistic attention network,” in *Proc. of European conf. on computer vision (ECCV)*, Milan, Italy, pp. 191–207, 2020.

[23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn and X. Zhai, “An image is worth 16 × 16 words: Transformers for image recognition at scale,” arXiv preprint arXiv: 2020.11929, 2010.

[24] J. Liang, J. Cao, G. Sun, K. Zhang, L. van Gool *et al.,* “SwinIR: Image restoration using swin transformer,” in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Virtual, pp. 1833–1844, 2021.

[25] N. Ahn, B. Kang and K. A. Sohn, “Fast, accurate, and lightweight super-resolution with cascading residual network,” in *Proc. of the European conf. on computer vision (ECCV)*, Munich, Germany, pp. 252–268, 2018.

[26] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan *et al.,* “Meta-sr: A magnification-arbitrary network for super-resolution,” in *Proc. of the IEEE/CVF conf. on computer vision and pattern recognition (CVPR)*, Long Beach, CA, USA, pp. 1575–1584, 2019.

[27] L. Wang, Y. Wang, Z. Lin, J. Yang, W. An *et al.,* “Learning a single network for scale-arbitrary super-resolution,” in *Proc. of the IEEE/CVF Int. Conf. On Computer Vision (ICCV)*, Virtual, pp. 4801–4810, 2021.

[28] W. Li, K. Zhou, L. Qi, L. Lu and J. Lu, “Best-buddy gans for highly detailed image super-resolution,” *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 36, pp. 1412–1420, 2022.

[29] E. Gothai, S. Bhatia, A. M. Alabdali, D. K. Sharma, B. R. Kondamudi *et al.,* “Design features of grocery product recognition using deep learning,” *Intelligent Automation & Soft Computing*, vol. 34, no. 2, pp. 1231–1246, 2022.

[30] M. A. B. Zuraimi and F. H. J. Zaman, “Vehicle detection and tracking using YOLO and DeepSort,” in *Proc. of 2021 11th IEEE Symp. on Computer Applications & Industrial Electronics (ISCAIE)*, Penang, Malaysia, pp. 23–29, 2021.

[31] R. Ravindran, M. J. Santora and M. M. Jamali, “Multi-object detection and tracking, based on DNN, for autonomous vehicles: A review,” *IEEE Sensors Journal*, vol. 21, no. 5, pp. 5668–5677, 2020.

[32] V. Premanand and D. Kumar, “Moving multi-object detection and tracking using MRNN and PS-KM models,” *Computer Systems Science and Engineering*, vol. 44, no. 2, pp. 1807–1821, 2023.

[33] Y. Rao, H. Mu, Z. Yang, W. Zheng and F. Wang, “B-PesNet: Smoothly propagating semantics for robust and reliable multi-scale object detection for secure systems,” *Computer Modeling in Engineering & Sciences*, vol. 132, no. 3, pp. 1039–1054, 2022.

[34] R. Girshick, J. Donahue, T. Darrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proc. of the 2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, pp. 580–587, 2014.

[35] K. He, X. Zhang, S. Ren and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[36] R. Girshick, “Fast R-CNN,” in *Proc. of the 2015 IEEE Int. Conf. on Computer Vision (ICCV)*, Boston, MA, USA, pp. 1440–1448, 2015.

[37] S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *Proc. of Advances in Neural Information Processing Systems (NIPS)*, Montreal, Canada, pp. 28, 2015.

[38] J. Jiang, D. Ergu, F. Liu, Y. Cai and B. Ma, “A review of yolo algorithm developments,” *Procedia Computer Science*, vol. 199, no. 11, pp. 1066–1073, 2022.

[39] J. Zhang, M. Huang, X. Jin and X. Li, “A real-time Chinese traffic sign detection algorithm based on modified yolov2,” *Algorithms*, vol. 10, no. 4, pp. 127, 2017.

[40] P. Skalski, *Make Sense*, 2019. [Online]. Available: https://github.com/SkalskiP/make-sense/

[41] E. Agustsson and R. Timofte, “Ntire 2017 challenge on single image super-resolution: Dataset and study,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Honolulu, HI, USA, 2017.