Blind Image Super-Resolution

Devashish Prasad
Department of Computer Science
Purdue University, West Lafayette, IN 47906
prasadd@purdue.edu
https://github.com/DevashishPrasad/superresolution

Abstract

Super-resolution (SR) has been studied and implemented since the 1990s because of its wide range of applications. It is used in Medical Imaging, CCTV surveillance, Gaming, and Astronomical Imaging, to name a few. Training practical SR models are typically challenging because these models tend to get biased toward training data distribution (type of degradation of low resolution (LR) images). Traditional SR models don't generalize well to real-world unknown test time images.

Recently, researchers have been paying more attention to making the SR models more robust such that they become invariant to the degradation process of the LR image input. It is known as the blind image SR task that aims to super-resolved LR images that result from an unknown degradation process and generate high resolution (HR) images. In this project, we present a detailed comparative analysis of two recent state-of-the-art Blind and one older but prominent Non-blind SR method. All three methods were originally trained (by their authors) in different training and testing environments. And so, these pre-trained models cannot be compared directly, which is the primary motivation of this project. To compare these models fairly, in this project, we carry out detailed experiments of training and evaluating these models in a common training and testing setting. We are the first to conduct such a study and compare these models directly, to the best of our knowledge. Code is available at https://github.com/DevashishPrasad/superresolution

1 Introduction

Image Super-Resolution (SR) refers to the task of enhancing the resolution of an image from low-resolution (LR) to high resolution (HR). Image super-resolution has been a topic of interest studied by researchers even before the deep learning era since the early 2000s [1,2,3,4,5]. After looking at the success of deep learning and CNNs in other computer vision tasks, many people have tried and improved CNN-based techniques a lot for image super-resolution tasks [6,7,8,9]. These studies assume that the LR image is a bicubic down-sampled (down-scaled using bicubic interpolation algorithm) version of HR. They train CNNs using these assumptions and use a dataset of HR (target y) images paired with their bicubic down-sampled LR (input x) images. The training and testing procedure of these simple CNN-based super-resolution models is shown in Fig 1. The performance of these models is estimated by comparing the generated image similarity with the ground truth HR image. And to compare two image similarities, the two widely used metrics are peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [10]. Acceptable values for wireless transmission PSNR quality loss are considered about 25 dB and more, and many SR models report an average of 26 to 29 dB on standard evaluation datasets of around 100 images. Despite these exciting improvements, these methods tend to fail in many real-world scenarios because of the bicubic down-sampling assumption (or a specific algorithm-based down-sampling assumption). The performance of SR models trained

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

in this way is limited to the kinds of inputs they are trained on, and it drops dramatically when tried on other kinds of inputs.

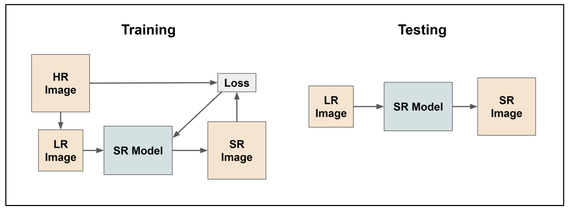

Figure 1: The figure shows traditional non-blind super-resolution models training and testing process. Note that loss (generally L1 loss) is calculated between HR Image ground truth and SR image prediction. And then, the gradients calculated using this loss are back-propagated through the SR Model network to train the weights

The inconsistency between the simplistic image down-sampling (image degradation) assumption of existing SR methods and the complex degradations of real-world images have led researchers to build degradation-aware SR models [11]. Over the years, many techniques have been proposed [36], each having its advantages and disadvantages. In this project, we choose to solve the blind SR problem using a specific type of method in which we construct and train a degradation estimation network along with our super-resolution model. The training and testing procedure of this type of blind super-resolution model is shown in Fig 2. The degradation estimation network predicts a unique representation that describes the degradation process. This representation is then concatenated with convolutional blocks of the super-resolution model. To train such a degradation estimation network, we use Equation 1, which can model (represent) any degradation process [13].

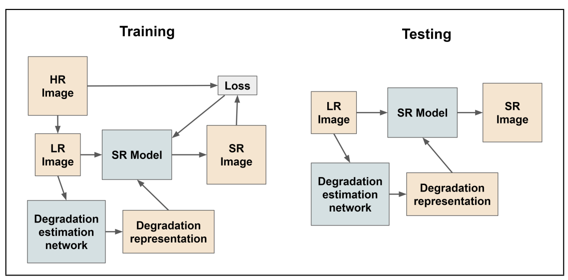

Figure 2: The figure shows the blind super-resolution model training and testing process. We will add a degradation estimation network which will take the LR image as input and produce a representation (mostly a vector or tensor) of the degradation of the LR image. We will train this network too. During the testing time we will use this trained degradation estimation network and make our SR model degradation aware.

Considering the inverse of the SR task, for a scale factor of s, the classical degradation model of SR assumes the LR image y is a blurred, decimated, and noisy version of an HR image x. Mathematically, this can be expressed by

$$y = (x \otimes k) \downarrow s + n \tag{1}$$

where $\otimes$ represents two-dimensional convolution of x with blur kernel k, $\downarrow$ s denotes the standard s-fold downsampler (bicubic in our case), and n is usually assumed to be additive white Gaussian noise (AWGN) specified by standard deviation (or noise level) σ. Using Equation 1, Zhang et al. [11] have been successful at super resolving LR images for which the degradation (k,s, and n) is already known. However, the blind image super-resolution task aims to recover the HR image from any LR image whose degradation is unknown to the SR model. In other words, a blind SR method first needs to estimate the degradation parameters (k, s, and n) and then super-resolve the image using these parameters. Note that, defining the degradation process like this explicitly, we assume that Equation 1 can represent any degradation. Some studies argue that Equation cannot represent all degradations, and they use other ways approach the problem and define the degradation process. However, we use Equation 1. because even if it cannot represent all degradations, it can certainly, represent many complex degradations and solving this problem will be a good direction forward.

2 Literature Review

In this project, we present ablation studies and comparative analysis of two recent state-of-the-art Blind SR methods KOALAnet [12] and UDRL [13], along with a prominent non-blind baseline EDSR [37]. In this section, we give an overview of the current blind SR research to justify our choice of these papers. Specifically, we propose a new taxonomy to categorize existing methods into different classes according to their ways of solving the Blind SR problem. And we choose these two Blind SR papers because these techniques are the most practical and promising.

All Image SR papers can be categorized into two groups, Non-blind SR and Blind SR. And all of the Blind SR papers can be categorized into two subgroups Domain Modeling and Degradation Modeling.

Domain Modeling SR papers [14, 15, 16, 17, 18] try to learn an SR model with unpaired images i.e. the model is trained on a dataset in which LR images are not derived from any HR images (i.e. LR mages occur naturally). In such a setting, ground truth HR images are unavailable, and only the target LR images are present. One assumption of this task is that all LR images have the same or similar degradation (they belong to the same degradation domain). There are various ways in which prior techniques have trained such models. But all of these models are trained to implicitly estimate the degradation using the LR images and then learn to super resolve the LR images using different HR images (or different HR-LR pairs of images) as reference. While test time, it is guaranteed that the model will be super resolving LR images of this same domain of training.

Degradation Modeling SR papers do not aim to learn the degradation implicitly. Instead, these papers aim to build a degradation estimator model that can estimate the degradation (based on explicit assumption like Equation 1) of any given LR image during test time. We categorize these papers into four subgroups 1) LR to Bicubic to SR, 2) Modified Loss Function, 3) Test Time Training, 4) Degradation Estimation. First, LR to Bicubic to SR techniques [19, 20] first try to build an image to image translation model that can translate any LR image into a Bicubic down-sampled look-alike image. And then, they use the Non-blind SR techniques to super resolve this bicubic look-alike image. So no matter what degradation LR images might have, they will be always converted to a bicubic look-alike images, and the traditional Non-blind SR will not struggle. Second, Modified Loss Function techniques [21, 22, 29] blame the the traditional L1 loss function of SR models as the culprit of poor performance in a blind setting. And so, these techniques train Non-blind SR models with very different and novel loss functions or architectures (mainly to get rid of L1 loss) such that the model becomes good at estimating blind degradations. Third, Test Time Training techniques [23, 24] train the SR models on various small patches extracted out of test time image and then super resolve the whole LR image after training. Finally, Degradation Estimation based techniques [25, 26, 27, 28, 12, 13] are the one we talk about in this paper. These techniques aim first at training a separate model that estimates the degradation (or a representation of the degradation) and then trains a degradation aware SR model that uses the output of the degradation estimation model while super resolving images.

Test Time Training-based models have an overhead of training at the test time that limits their applications in real-world. LR to Bicubic to SR and Modified loss functions techniques don't attain higher metric scores and struggle with a diverse set of degradations. And this is the reason researchers have been paying the most attention to the degradation estimation-based techniques, out of which we have chosen the two state-of-the-art methods [12] and [13].

3 Models

The EDSR [37] is a Non-blind model having a single SR network. While, Both [12] and [13] train two networks, the first network estimates the degradation of the LR image, and the second network uses this degradation estimation to super resolve the LR. Both techniques [12, 13] use Equation 1 as the base assumption to model degradation.

3.1 EDSR

The Non-blind EDSR model [37] architecture is made by stacking 32 Residual blocks one after the other. Each block has the same shape and size with 3x3 filters and 256 features or channels. In the end, EDSR has a pixel shuffle upsampling layer after all Residual blocks. It rearranges elements in a tensor of shape (C $\times r^2$, H, W) to a tensor of shape (C, H $\times$ r, W $\times r$), where C is number of channels, r is an upscale factor, H is height and W is width.

The original training strategy for this model involved training on patches of size 48x48 extracted from LR images of the respective HR images. The authors used L1 loss and two augmentation strategies random horizontal flips and random 90 rotations.

3.2 UDRL

The blind UDRL [13] model has two networks. Its first network (degradation estimation network) learns to predict the representation of the degradation using a contrastive learning-based technique called MoCo [38]. It has a simple architecture with just 6 convolutional layers with kernel size 3x3. The output vector of this first vector is concatenated in certain CNN blocks of the second network (SR network) while super-resolving the LR image. These blocks are called Degradation Aware (DA) blocks. Each DA block performs a non-trivial operation using two full-connected (FC) layers and a reshape layer to concatenate the representation vector. The SR network consists of 5 residual groups, with each group comprising of 5 DA blocks. Similar to EDSR, the SR network has a pixel shuffle upsampling layer at the end after all residual groups and DA blocks.

The original training strategy involved training UDRL in different settings. It was first trained on noise-free degradations with isotropic Gaussian kernels with different ranges of kernel width. Then, the network was trained on more general degradations with anisotropic Gaussian kernels and noises. Again with different ranges of the covariance matrix. The traning strategy involves training the degradation encoder for 100 epochs under a contrastive learning strategy. And then, the whole network (encoder + SR model) is trained for 500 epochs jointly end to end. The overall loss function is defined as L = L1 loss + Contrastive loss.

3.3 KOALAnet

KOALAnet [12] has a total of three networks. Its first network takes the LR image as input and predicts the degradation kernels (which were used for degrading the image) for it's each pixel location. The loss function for this first network has two parts. First, it computes the loss by comparing the predicted kernels against the ground truth kernels. Second, it upsamples the LR image using the predicted kernels and compares the generated image with the ground truth HR image. These predicted degradation kernels for the LR input image also act as a degradation representation. These kernels (tensor) are fed into the second network of KOALAnet which is a relatively small network with few convolutional layers. The resulting transformed output of this second network is concatenated with the CNN blocks (called degradation-aware convolutional blocks) of the third network. The third network is a big SR network that super-resolves the LR image using several degradation-aware convolutional blocks, followed by two different branches. Both branches take the same input (output of degradation-aware convolutional blocks). The first branch predicts the inverse kernels to upsample the LR image, and the second branch has the pixel shuffle layer as seen in UDRL and EDSR. The resulting SR is created by concatenating the outputs of these two layers. This third network is trained using the loss function that compares the HR image generated by upsampling the LR image using these predicted kernels.

The original training strategy involved three phases (i) the downsampling network (first network) is pre-trained with LR input (ii) the upsampling network (third network) is pre-trained with LR input by replacing all KOALA modules with ResBlocks; (iii) the whole framework (KOALAnet) including the KOALA modules (on the pre-trained ResBlocks) is jointly optimized. KOALAnet was trained using the complex degradations with anisotropic Gaussian kernels but without noises. Again with different ranges of the covariance matrix whose values are sampled randomly from a probability distribution.

Table 1: PSNR and SSIM of pre-trained models on 5 standard benchmark datasets. (upscaling factor x4)

Method	Set 5	Set 14	BSD 100	Urban 100	Manga 109	Div2K Val 100
Bicubic	24.86/0.7105	23.20/0.6240	23.86/0.6030	20.67/0.5774	22.09/0.7183	25.71/0.7184
EDSR	25.04/0.7239	23.49/0.6343	24.08/0.6116	21.06/0.6014	22.76/0.7407	26.10/ 0.7301
UDRL	27.34/0.7999	24.78/0.6922	25.02/0.6711	22.37/0.6815	25.43/0.8236	27.20/0.7756
KOALAnet	28.23/0.8337	25.31/0.7105	25.54/0.6873	23.17/0.7137	26.52/0.8488	27.96/0.7939

4 Experiments

All three models used 800 training images in DIV2K [30] as the training set, and included five benchmark datasets (Set5 [32], Set14 [33], B100 [34], and Urban100 [35], DIV2K Val[30]) for evaluation. [13] use extra 2650 training images from Flickr2K [31] dataset. All these datasets are used heavily by Non-blind SR techniques. The authors of the DIV2K dataset have released LR-HR pairs (LR generated using bicubic) for various scales, and typically every paper uses x2 and x4 upsampling scales. In a Blind SR setting, the papers ignore the bicubic LRs of DIV2K and generate their own versions of LR images while training. They use some probability distribution to sample Gaussian kernels and noise parameters and then use these parameters to produce LR images (using Equation 1) from their respective HR images. Blind SR models are trained on such a probabilistically generated dataset in which the SR model (upsampler) does not know what these degradation parameters are.

The problem is that the authors have not used a common benchmark dataset to train and evaluate their models. They have used their own versions of datasets for training and evaluation. Their crucial training factors like training data size, degradation modeling to create LR images, hyper-parameter tuning, evaluation metrics, etc. vary a lot. Every Blind SR technique reports PSNR and SSIM on the same evaluation datasets for x2 and x4 scales, but again the evaluation scheme differs because of different degradation modeling parameters. And so, we present the following experiment that does the comparative analysis of these three models on a common train and test settings.

4.1 Evaluating the pre-trained models

For the common and fair evaluation, we use a test set published by [12] which was generated using five benchmark datasets using different degradation parameters (of Equation 1) for different images randomly. We find this dataset the most appropriate because it was created using a complex degradation kernel of type an-isotropic. Such a difficult dataset benchmark will test all the models to their limit. We only use a 4x upscaling factor as it is more challenging than 2x, and the model needs to really perform well to upscale an image four times. Lastly, we use consistent metrics of PSNR and SSIM over whole images (instead of considering just a particular channel) across all models.

On this common benchmark dataset, we first test the three models using their pre-trained weights released by the authors (GitHub codes). All Github released codes are implemented using different and older versions of Deep Learning frameworks (EDSR=OpenCV 3.X, UDRL=Pytorch 1.1.0, and KOALAnet=TensorFlow 1.13). We evaluate these pre-trained model on evaluation dataset of [12] and table 1. shows the results of our evaluation.

4.2 Reproducing and training models from scratch

The previous pre-trained Github code models-based evaluation was not fair because each model was trained and evaluated differently. So, to train all models under the same scheme and dataset, we reproduce EDSR and UDRL from scratch in Pytorch and train them using a common dataset. Unfortunately, KOALAnet was written in TensorFlow 1.X, and because there were lots of complications in the code, we were not able to reproduce it in Pytorch. But, the authors did publish the training and evaluation code, so we use the same to train the model from scratch and change the evaluation code according to our dataset and metrics. As we saw in section 3, all models had their own different original training strategies. We remove this inconsistency by using a common training strategy i.e same dataset, degradation parameters (same training data distribution), augmentations, a learning rate scheduler, etc. across all models. We also keep the number of parameters and number of epochs about the same for all models. Table 2. shows the results of our evaluation of these models trained from scratch.

Table 2: PSNR and SSIM of models trained from scratch on 5 standard benchmark datasets. (upscaling factor x4)

Method	No Params	No Epochs	Set 5	Set 14	BSD 100	Urban 100	Manga 109	Div2K Val 100
Bicubic	NA	NA	24.86/0.7105	23.20/0.6240	23.86/0.6030	20.67/0.5774	22.09/0.7183	25.71/0.7184
EDSR+	5,487,433	300	25.68/0.7393	23.02/0.6347	23.82/0.6287	20.94/0.6041	22.64/0.7303	25.74/0.73060
UDRL	5,969,283	100+200	20.70/0.5381	19.99/0.4913	21.50/0.5007	19.22/0.4654	18.26/0.5337	22.65/0.5980
KOALAnet	6,545,416	50+50+110	25.72/0.7752	23.88/0.6721	24.39/0.6529	21.60/0.6515	23.40/0.7839	26.22/0.7592

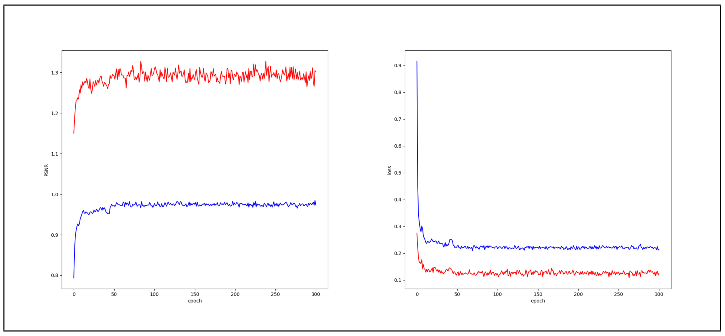

Figure 3: EDSR - Left: PSNR vs Epochs, Right: Loss vs Epochs.

EDSR was trained in the blind setting, meaning EDSR was presented LR images with different degradation parameters each time. We call this model EDSR+. Training EDSR was very straightforward because it follows the exact same process as Fig 1. The only difference is that LR images are generated from HR images with random degradations every time while training. We train EDSR for total 300 epochs.

UDRL has two components, the degradation estimation network (Encoder) and the SR model. The encoder is trained first for 100 epochs under a specific contrastive learning setting. And after 100 epochs, we start training both the encoder and SR model together for more than 200 epochs (a total of 300 epochs). We simply add the losses (contrastive and L1) and let the gradient back-propagate through both networks (end-to-end training). Just like EDSR+, UDRL was also trained under random degradations strategy.

KOALAnet's training involved three stages. First, we train the downsampling network (first network) for 50 epochs. Then we train the upsampling network (third network) for 50 epochs. And last, we train the whole framework (KOALAnet) for 110 epochs (a total of 210 epochs). Just like EDSR+ and UDRL, KOALAnet was also trained under random degradations strategy.

We also present the learning curves of all three models. Figure 3 shows training curves for EDSR while Figure 4 shows training curves for UDRL, and Figure 5, 6, and 7 shows training curves of phase 1, phase 2, and phase 3 of KOALAnet. The Red color represents train set performance or loss curve, and the Blue color represents the validation set performance or loss curve. We do not present the training curve for KOALANet because of some issues in the code. But validation curves show us evidence of successful training in KOALAnet. We discuss these curves in detail in next section.

5 Discussion and Analysis

As the training and evaluation methodologies were the same for all the models, we can now derive conclusions we can rely more on. Following is our analysis of training procedure and final performance.

We can see that EDSR's performance gets plateaued after a certain number of epochs. It shows that model has reached its limit and cannot improve further. But, it performs almost the same or slightly better than the bicubic baseline, and it further suggests that EDSR do possess some capability to solve blind SR task. Perhaps, increasing the model architecture will help, but this also proves that the Blind SR task can be efficiently solved by building more specific architecture like the other two approaches.

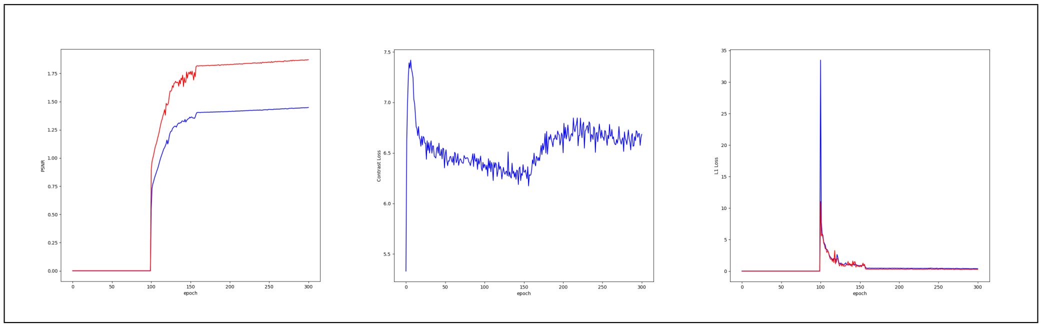

Figure 4: Left: UDRL SR Model - PSNR vs Epochs, Middle: UDRL Encoder - Loss vs Epochs, Right: UDRL SR Model - Loss vs Epochs

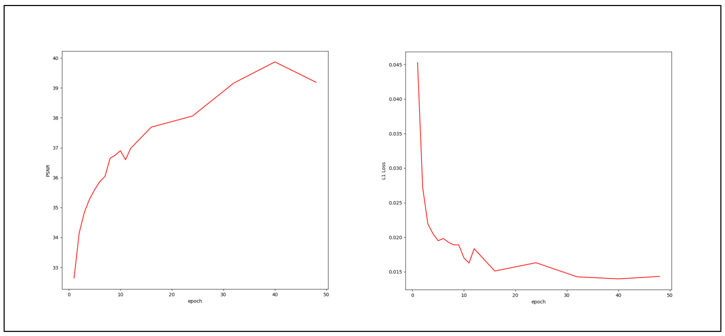

Figure 5: KOALAnet Encoder (network 1) - Left: PSNR vs Epochs, Right: Loss vs Epochs

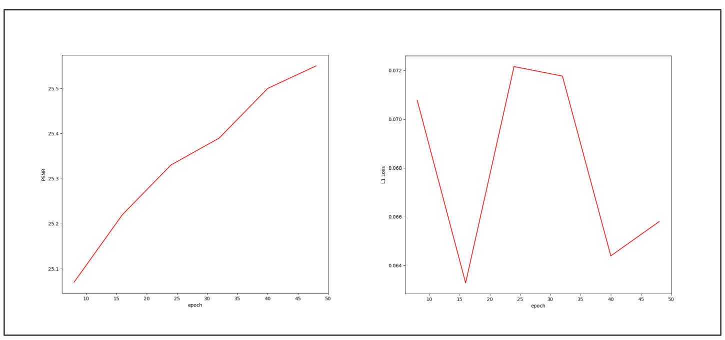

Figure 6: KOALAnet SR Model (network 3) without Encoder (network 1) - Left: PSNR vs Epochs, Right: Loss vs Epochs

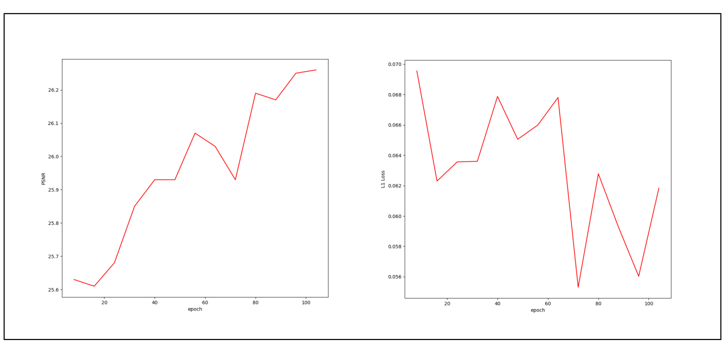

Figure 7: Whole KOALAnet - Left: PSNR vs Epochs, Right: Loss vs Epochs

We can also see a similar trend of plateauing while training UDRL. However, UDRL's performance is not completely plateau, and it keeps growing very slowly. We see an abrupt rise in UDRL encoder loss near epoch 160, probably because of a change in learning rate (done by LR Scheduler). The original UDRL model was trained on a much bigger dataset, and this tells us that either contrastive learning needs a lot of training time or needs a bigger training dataset.

But, by looking at KOALAnet curves, we can see that the network did not reach a saturation point yet, and the performance will increase if we train for more epochs. Note that in the benchmark KOALAnet was trained for the least number of epochs, but it still performs the best. It clearly indicates that KOALAnet architecture and methodology are very efficient and are the most suitable for solving the Blind Super resolution task when compared with the other two techniques.

As the authors of KOALAnet claim that KOALAnet can handle spatially variant degraded LR images because the downsampler predicts degradation kernels for each pixel location of the LR image, and the upsampler predicts inverse degradation kernels for each pixel location of the LR image. UDRL fails to consider spatially variant degradations and assumes that the whole LR image has the same degradation. However, even if we have trained and evaluated both models using the same degradation on the whole LR image and not spatially variant, we still see KOALAnet perform better. It tells us KOALAnet inherently has a better capability at estimating degradations from LR images. It could be the reason KOALAnet performs better than UDRL.

6 Conclusion

In this report, we introduced the blind image super-resolution task and presented a degradation modeling-based strategy to train the CNN-based deep learning models. We trained and evaluated three models namely EDSR, UDRL, and KOALAnet, under the exact same environment across all models. We draw several useful conclusions from our experiments. We showed that KOALAnet learns quickly and attains the best performance (PSNR and SSIM) as compared to the other two approaches. Our experiments support this claim by giving evidence about KOALAnet being the most effective for Blind SR task. For our future work, we plan to train and test the models with more training epochs, different model sizes, and more complex noisy degradations. It would give us even more clarity about which model performs the best and why is that the case.

7 References

[1] J. Allebach and P. W. Wong. Edge-directed interpolation. In *ICIP 1996*

[2] X. Li and M. T. Orchard. New edge-directed interpolation. *IEEE Transactions on ImageProcessing*, 10(10):1521–1527, 2001.

[3] L.Zhang and X.Wu. An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Transactions on Image Processing*, 15(8):2226–2238, 2006.

[4] Y.-W. Tai, S. Liu, M. S. Brown, and S. Lin. Super resolution using edge prior and single image detail synthesis. In *CVPR 2010*.

[5] J. Sun, Z. Xu, and H.-Y. Shum. Image super-resolution using gradient profile prior. In *CVPR 2008*.

[6] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *ECCV 2014*

[7] C. Dong, C. C. Loy, and X. Tang. Accelerating the superresolution convolutional neural network. In *ECCV 2016*

[8] J. Kim, J. Kwon Lee, and K. M. Lee. Accurate image super resolution using very deep convolutional networks. In *CVPR 2016*.

[9] J. Kim, J. Kwon Lee, and K. M. Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR 2016*.

[10] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13(4), 600–612 (2004)

[11] Zhang, Kai and Van Gool, Luc and Timofte, Radu. Deep unfolding network for image super-resolution, *IEEE Conference on Computer Vision and Pattern Recognition*, 3217–3226, 2020

[12] Kim, Soo Ye and Sim, Hyeonjun and Kim, Munchur, KOALAnet: Blind Super-Resolution using Kernel-Oriented Adaptive Local Adjustment, Proceedings of *the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages=10611–10620, 2021

[13] Wang, Longguang and Wang, Yingqian and Dong, Xiaoyu and Xu, Qingyu and Yang, Jungang and An, Wei and Guo, Yulan, Unsupervised Degradation Representation Learning for Blind Super-Resolution, *CVPR, 2021*

[14] Bulat, Adrian and Yang, Jing and Tzimiropoulos, Georgios, To learn image super-resolution, use a GAN to learn how to do image degradation first, *Proceedings of the European Conference on Computer Vision (ECCV)*, 185–200, 2018

[15] Yong Guo, Jian Chen, Jingdong Wang, Qi Chen, Jiezhang Cao, Zeshuai Deng, Yanwu Xu, Mingkui Tan. Closed-loop Matters: Dual Regression Networks for Single Image Super-Resolution. *CVPR, 2020*

[16] Maeda, Shunta. "Unpaired Image Super-Resolution using Pseudo-Supervision." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

[17] Wei, Yunxuan and Gu, Shuhang and Li, Yawei and Timofte, Radu and Jin, Longcun and Song, Hengjie, Unsupervised Real-World Image Super Resolution via Domain-Distance Aware Training, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2021

[18] Wei Wang, Haochen Zhang, Zehuan Yuan, Changhu Wang. Unsupervised Real-World Super-Resolution: A Domain Adaptation Perspective. *ICCV 2021*

[19] Shady Abu Hussein, Tom Tirer, Raja Giryes. Correction Filter for Single Image Super-Resolution: Robustifying Off-the-Shelf Deep Super-Resolvers *CVPR 2020*

[20] Mohammad Saeed Rad, Thomas Yu, Claudiu Musat, Hazim Kemal Ekenel, Behzad Bozorgtabar, Jean-Philippe Thiran. Benefiting from Bicubically Down-Sampled Images for Learning Real-World Image Super-Resolution. *WACV 2021*

[21] Zheng Hui, Jie Li, Xiumei Wang, Xinbo Gao; *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2093-2102

[22] Younghyun Jo, Seoung Wug Oh, Peter Vajda, Seon Joo Kim; *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 16236-16245

[23] Mohammad Emad, Maurice Peemen, Henk Corporaal; *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 1630-1639

[24] Jingyun Liang, Kai Zhang, Shuhang Gu, Luc Van Gool, Radu Timofte. Flow-based Kernel Prior with Application to Blind Super-Resolution. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[25] Liang, Jingyun and Sun, Guolei and Zhang, Kai and Van Gool, Luc and Timofte, Radu, Mutual Affine Network for Spatially Variant Kernel Estimation in Blind Image Super-Resolution, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pages=4096-4105

[26] Zhengxiong Luo, Yan Huang, Shang Li, Liang Wang, Tieniu Tan. Unfolding the Alternating Optimization for Blind Super Resolution. *Conference on Neural Information Processing Systems (NeurIPS) 2020*

[27] Liangbin Xie, Xintao Wang, Chao Dong, Zhongang Qi, Ying Shan. Finding Discriminative Filters for Specific Degradations in Blind Super-Resolution. *Conference on Neural Information Processing Systems (NeurIPS) 2020*

[28] Guangpin Tao, Xiaozhong Ji, Wenzhuo Wang, Shuo Chen, Chuming Lin, Yun Cao, Tong Lu, Donghao Luo, Ying Tai. Spectrum-to-Kernel Translation for Accurate Blind Image Super-Resolution. *Conference on Neural Information Processing Systems (NeurIPS) 2020*

[29] Qian Ning, Weisheng Dong, Xin Li, Jinjian Wu, GUANGMING Shi. Uncertainty-Driven Loss for Single Image Super-Resolution. *Conference on Neural Information Processing Systems (NeurIPS) 2020*

[30] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. *In CVPRW, pages 1122–1131, 2017.*

[31] Radu Timofte, Eirikur Agustsson, Luc Van Gool, MingHsuan Yang, Lei Zhang, and Bee Lim et al. NTIRE 2017 challenge on single image super-resolution: Methods and results. *In CVPRW, pages 1110–1121, 2017.*

[32] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie-Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In BMVC, pages 1–10, 2012

[33] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. *In International Conference on Curves and Surfaces, volume 6920, pages 711–730, 2010*

[34] David Martin, Charless Fowlkes, Doron Tal, Jitendra Malik, et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *In ICCV, 2001*

[35] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. *In CVPR, pages 5197–5206, 2015*.

[36] Liu, Anran and Liu, Yihao and Gu, Jinjin and Qiao, Yu and Dong, Chao. Blind Image Super-Resolution: A Survey and Beyond. *https://arxiv.org/abs/2107.03055*.

[37] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced Deep Residual Networks for Single Image Super-Resolution. *2nd NTIRE: New Trends in Image Restoration and Enhancement workshop and challenge on image super-resolution in conjunction with CVPR 2017*

[38] Kaiming He and Haoqi Fan and Yuxin Wu and Saining Xie and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. *arXiv preprint arXiv:1911.05722. 2019*.