

Project: Super-Resolution

Name: Achint Chaudhary, D Awadhesh Tanwar

Problem Statement: Enhancing low-resolution images using deep neural networks to produce high-resolution images.

Datasets: Training (DIV2K, Flickr2K, CLIC-mobile, CLIC-professional), Benchmarking (PIRM, BSDS100, Urban100, Manga109, Set14, Set5 and validation sets from training) are datasets used in the literature.

Abstract

Single Image Super-Resolution (SISR) is the task of constructing high resolution from a low-resolution image, which finds its usage in satellite, security, medical imaging. Deep learning-based solutions in recent years have outperformed various classical techniques. A lot of literature work, neural architectural innovations and datasets are released in various challenges. This project aims to analyse both qualitative and quantitative aspects of various methods and datasets in the past few years, that have contributed towards the state of the art in SISR.

We have proposed a novel image sampling algorithm for selecting images during training, for better convergence. We have used popular CNN architecture: SRResNet, for verification of our sampling algorithm. In the process, we built a framework for easy implementation of CNN based techniques for low-level vision applications like Super-Resolution, De-noising, and Compression Artefacts Removal.

Description

Classical interpolation techniques like nearest or bicubic upsampling are non-iterative approaches in nature but suffer in term of quality of results.

Various CNN based models for SISR can be classified in to two categories: type(i) model having input as Upsampled interpolated image by classical methods (VDSR, DRRN, OA-DNN); type(ii) models that take LR image and extract features in LR space itself, and later use a learnable upsampling kernel (SRResNet, EDSR). We have used type(ii) models because type(i) are highly memory & time consuming for both training & testing, also they have not converged under given training settings in their literature.

Our input images can be of different formats, and different dimensions, which is handled by taking fixed size patches from the images.

Like Mini-batch gradient descent, which takes batches in memory, we have taken batches of images on disk, this allows us to work in memory-constrained systems and to sample over entire available training dataset. With multiple large-sized, high-quality datasets, it becomes more important which data set to choose from & if possible which images from specific datasets suit more for training. Recently, E-SRGAN have combined DIV2K & Flickr2K for obtaining best results at the cost of more training resources. This approach will saturate in case of very large datasets.

For the above reasons, we have proposed a novel sampling algorithm for selection of images.

Experiment 0 (Model Selection)

Tried various of type(i) & type(ii) models, SRResNet from type(ii) is chosen for further experiments, as it provides best results, and is fastest among all models taken under consideration (type(i) were unfeasibly slow for training).

Experiment 1 (Data-Set Ranking)

We trained separate instances of SRResNet, for each training dataset. Our performance metric is Peak Signal to Noise Ratio (PSNR).

PSNR distribution using bicubic for training sets is plotted (**Fig 1.**). Order of peak values in PSNR distribution for training sets is DIV2K < Flickr2K < CLIC-P < CLIC-M. This shows that CNN models have more opportunity to learn in reverse order, which is also empirically verified in results.

Distribution of PSNR Gain on various test sets, obtained from CNN trained on DIV2K is plotted in (Fig 2.). This shows that datasets having fewer details like Manga109, Set5 have more PSNR gain as compared to other realistic ones.

We compared the performance of each model instance on test sets and obtained PSNR gain. (**Fig 3**). Testing on independent data set from the recent challenges also supports our claim that DIV2K is best among all for learning

Results of mean PSNR Gain for all instances and all test sets are in Table 1.

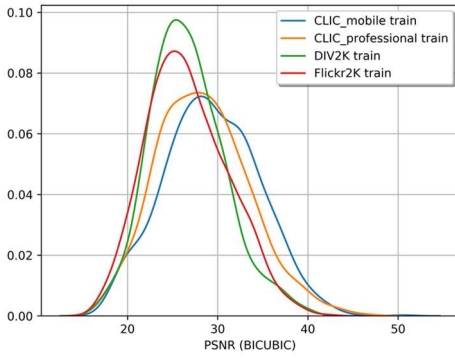


Figure 1

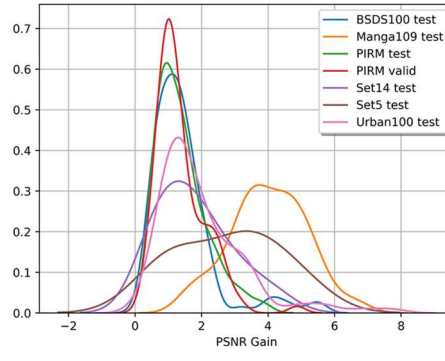


Figure 2

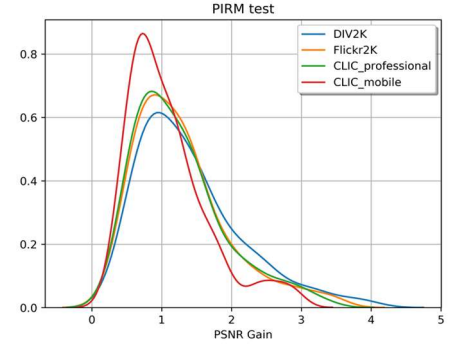


Figure 3

Dataset	DIV2K	Flickr2K	CLIC_professional	CLIC_mobile
CLIC_mobile valid	1.43	1.28	1.34	1.16
CLIC_professional valid	1.64	0.88	1.49	1.20
DIV2K valid	1.55	1.11	1.27	1.12
BSDS100 test	1.73	1.39	1.44	1.32
Manga109 test	4.01	3.75	3.39	3.03
PIRM valid	1.40	1.27	1.25	1.08
PIRM test	1.37	1.24	1.19	1.02
Set5 test	1.79	1.65	1.57	1.38
Set14 test	2.77	2.33	2.45	1.98
Urban100 test	2.00	1.86	1.78	1.57

Table 1. (Performance of different trained instances of SRResNet on various test sets)

Experiment 2 (Data-Set Sampling)

In the above experiment, we have ranked the data sets for the best performance. This motivates us to rank images while training for faster model convergence. Since we have disk batching to sample images. We propose a novel sampling algorithm that ranks images for disk batching on-the-fly.

Algorithm: Let N be number of images in the entire training dataset

Step 0: Assign each image, a uniform probability of $1/N$

While (Model Converges)

Step 1: Take sample of K images, proportional to present probabilities of images

Step 2: Calculate the PSNR gain over these images

Step 3: Update probabilities of K images, (using sample distribution of PSNR gain)

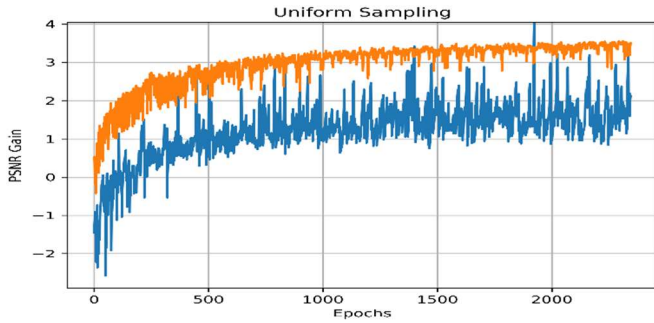


Figure 4

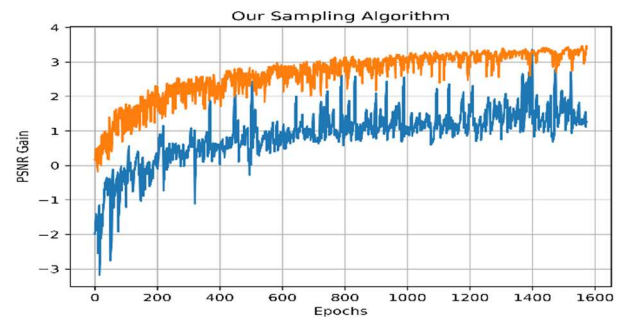


Figure 5

Performance of our sampling algorithm over uniform sampling is shown in Fig 4 & 5. We observe our algorithm will help training to converge faster than uniform approach (1600 over 2500). Both these observations favour our algorithm.

Conclusion & Take-away

1. Analysed & Benchmarked various CNN models from literature.
2. Characteristics of various training & performance benchmark datasets are utilised for finding best training sets.
3. Proposed & empirically verified an Image sampling algorithm for faster convergence of the model.

Future Work

1. Initial probability values can be assigned based on Bicubic Up-sampling rather than Uniform assignment.
2. Usage of Dataset ranking & sampling approaches can be explored in other domains like Compression Artefacts Removal, De-noising