Super-Resolution Neural Operator

Parviz Karimov

MIPT, 2024

May 5, 2024

Motivation

Most DNNs are developed in the configuration of single scaling factors, which cannot be used in scenarios requiring arbitrary SR factors. Recently, implicit neural functions (INF) have been proposed to represent images in arbitrary resolution, and paving a feasible way for continuous SR. To share knowledge across instances instead of fitting individual functions for each signal, encoder-based methods are proposed to retrieve latent codes for each signal, and then a decoding MLP is shared by all the instances to generate the required output, where both the coordinates and the corresponding latent codes are taken as input. However, the point-wise behavior of MLP in the spatial dimensions results in limited performance when decoding various objects, particularly for high-frequency components

Problem Statement

Consider the following PDE:

$$(L_a u)(x) = f(x), x \in D$$

$$u(x) = 0, x \in \partial D,$$

where $u : D \to \mathbb{R}^{d_u}$ is the solution function residing in the Banach space $\mathcal{U}$, and $L : \mathcal{A} \to L(\mathcal{U}, \mathcal{U}^*)$ is an operator-valued functional that maps the coefficient function $a \in \mathcal{A}$ of the PDE to $f \in \mathcal{U}^*$.
Neural operator (NO) seeks a feasible operator $\mathcal{G} : \mathcal{A} \to \mathcal{U}, a \mapsto u$, directly mapping the coefficient to the solution within an acceptable tolerance.

The infinitely dimensional operator learning problem $\mathcal{G} \leftarrow \mathcal{G}_\theta$ is associated with the empirical-risk minimization problem.

$$\min_\theta \mathbb{E}_{a\sim\mu} ||\mathcal{G}(a) - \mathcal{G}_\theta(a)||_{\mathcal{U}} \approx \min_\theta \frac{1}{N} \sum_j ||u^{(j)} - \mathcal{G}_\theta(a^{(j)})||_{\mathcal{U}}$$

SRNO Problem Statement

Let $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}})$ be a Hilbert space equipped with the inner-product structure, which is continuously embedded in the space of continuous functions $C^0(\Omega)$, with $\Omega \subset \mathbb{R}^2$ a bounded domain. An image is defined as a vector-valued function $\mathcal{H} \ni f : \Omega \to \mathbb{R}^3$. Assume we can access the function values of f at the coordinates $\{x_i\}_{i=1}^{n_h} \subset \Omega$ with the biggest discretization size h.

The goal is to learn a super-resolution neural operator between two Hilbert spaces with different resolutions: $\mathcal{S}_\theta : \mathcal{H} \supset \mathcal{H}(\Omega_{h_c}) \to \mathcal{H}(\Omega_{h_f}) \subset \mathcal{H}$, where h_c, h_f denotes the coarse and the fine grid sizes, respectively. Given N function pairs $\{a^{(k)}, u^{(k)}\}_{k=1}^N$, where $a^{(k)} \in \mathcal{H}(\Omega_{h_c})$ and $u^{(k)} \in \mathcal{H}(\Omega_{h_f})$. SRNO parameterized by θ can be solved through the associated empirical-risk minimization problem:

$$\min_\theta \frac{1}{N} \sum_{k=1}^{N} \| u_{h_f}^{(k)} - \mathcal{S}_\theta(a_{h_c}^{(k)}) \|_{\mathcal{H}}.$$

Overview

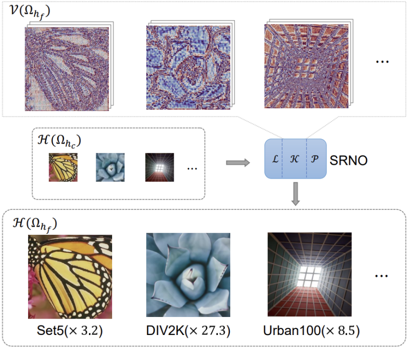

Figure 1. **Overview of Super-Resolution Neural operator (SRNO).** SRNO is composed of three parts, $\mathcal{L}$ (Lifting), $\mathcal{K}$ (kernel integrals) and $\mathcal{P}$ (Projection), which perform consecutively to learn mappings between approximation spaces $\mathcal{H}(\Omega_{h_c})$ and $\mathcal{H}(\Omega_{h_f})$ associated with grid sizes h_c and h_f, respectively. The key component, $\mathcal{K}$, uses test functions in the latent Hilbert space $\mathcal{V}(\Omega_{h_f})$ to seek instance-specific basis functions.

Overview

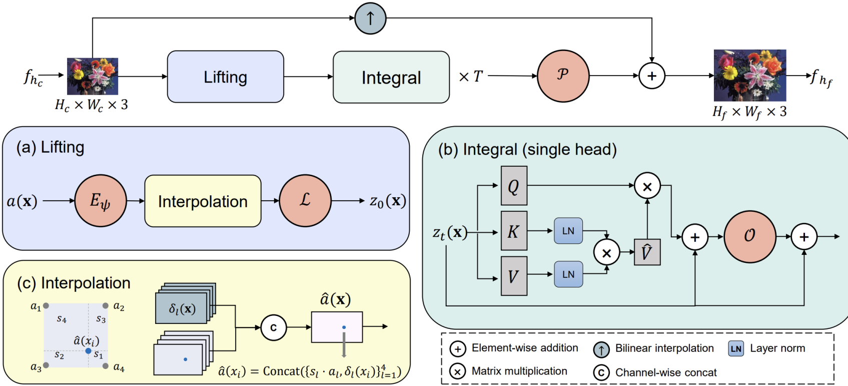

Figure 2. **Super-resolution neural operator (SRNO) architecture for continuous SR.** The input LR image f_{h_c} undergoes three phases to output the HR image f_{h_f} with the specified resolution: (a) Lifting the LR pixel values $a(\mathbf{x})$ on the set of coordinates $\mathbf{x} = \{x_i\}_{i=1}^{n_{h_f}}$ to a higher dimensional feature space by a CNN-based encoder E_ψ, constructing the latent representation $\hat{a}(x)$, and linearly transforming into the first layer's input $z_0(\mathbf{x})$. (b) kernel integrals composed of T layers of Galerkin-type attention, and (c) finally project to the RGB space.

Overview

As a whole, the process of getting HR image as a result of SRNO is the following:

$$
\begin{aligned}
z_0(x) &= \mathcal{L}(x, a(x)), \\
z_{t+1}(x) &= z_t(x) + \mathcal{O}((\mathcal{K}_t(z_t))(x) + z_t(x)), \\
u(x) &= \mathcal{P}(z_T(x)),
\end{aligned}
$$

where $\mathcal{L} : \mathbb{R}^{d_a+d} \to \mathbb{R}^{d_z}$, and $\mathcal{P} : \mathbb{R}^{d_z} \to \mathbb{R}^{d_u}$ are the local lifting and projection functions respectively , mapping the input a to its first layer hidden representation z_0 and the last layer hidden representation z_T back to the output function u. $W : \mathbb{R}^{d_z} \to \mathbb{R}^{d_z}$ is a point-wise linear transformation, and $\sigma : \mathbb{R}^{d_z} \to \mathbb{R}^{d_z}$ is the nonlinear activation function.

Quantitative Results

Method	In-distribution			Out-of-distribution				
	$\times 2$	$\times 3$	$\times 4$	$\times 6$	$\times 12$	$\times 18$	$\times 24$	$\times 30$
Bicubic	31.01	28.22	26.66	24.82	22.27	21.00	20.19	19.59
EDSR-baseline	34.55	30.90	28.94	-	-	-	-	-
EDSR-baseline-MetaSR	34.64	30.93	28.92	26.61	23.55	22.03	21.06	20.37
EDSR-baseline-LIIF	34.67	30.96	29.00	26.75	23.71	22.17	21.18	20.48
EDSR-baseline-LTE	34.72	31.02	29.04	26.81	23.78	22.23	21.24	20.53
EDSR-baseline-SRNO (ours)	**34.85**	**31.11**	**29.16**	**26.90**	**23.84**	**22.29**	**21.27**	**20.56**
RDN-MetaSR	35.00	31.27	29.25	26.88	23.73	22.18	21.17	20.47
RDN-LIIF	34.99	31.26	29.27	26.99	23.89	22.34	21.31	20.59
RDN-LTE	35.04	31.32	29.33	27.04	23.95	22.40	21.36	20.64
RDN-SRNO (ours)	**35.16**	**31.42**	**29.42**	**27.12**	**24.03**	**22.46**	**21.41**	**20.68**

Quantitative comparison on DIV2K validation set (PSNR (dB)). The best performance are bolded. EDSR-baseline trains separate models for the three in-distribution scales. The rest methods use a single model for all scales, and are trained with continuous random scales uniformly sampled in $\times 1$–$\times 4$.

Qualititative Results

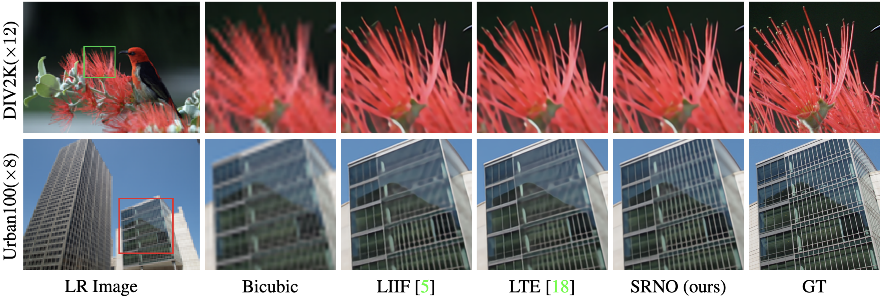

Figure 3. **Visual comparison on other zero-shot SR**. The boxes in the first column indicate the areas that the close-ups on the right display. All methods are trained with continuous random scales in $\times 1$–$\times 4$ and tested on $\times 8 / \times 12$ to evaluate the generalization capability to unknown scaling factors. RDN is used as the encoder for all methods.

Qualititative Results

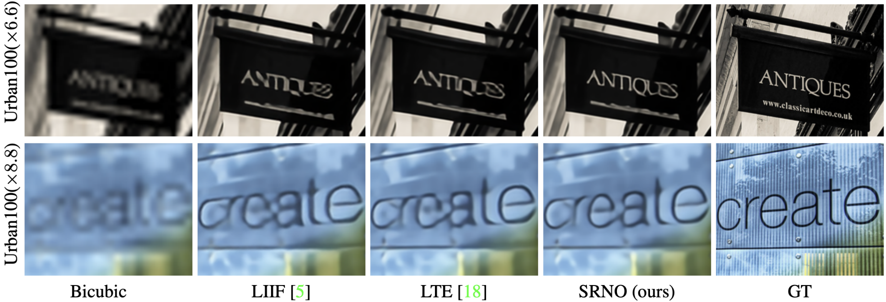

Figure 4. **Visual comparison on non-integer scales**. All methods use RDN as the encoder and are trained with continuous random scales in $\times 1$–$\times 4$.

Literature

1. **Main article** Super-Resolution Neural Operator.