

# Materials: Scalable Super-Resolution Neural Operator

Anonymous Authors

In this supplementary, we first provide more details on some concepts mentioned in the main document, which are not only helpful for understanding the main idea but also important in the algorithmic implementation. Then, we present another piece of evidence to illustrate the effectiveness of our method. Finally, we present more visualization results on SSRNO.

## A ADDITIONAL TECHNICAL DETAILS

### A.1 Layer grouping for dimensionality matching

A group of layers refers to a collection of parameter matrices following the constraint of sharing the same dimensionality in the channel dimension. For convolutional neural network (CNN) and fully connected neural network (FCNN), the constraint can occur between adjacent layers or within residual skip connections in residual layers. For the transformer, the relationship can be found in the Q/K/V multiplication calculations as well as the layer-norm operations, *etc.* For an example of adjacent layers in CNN, the layers can be formulated as:

$$\begin{aligned} Z_0 &= x \\ Z_{n+1}(x) &= \sigma(\text{CONV}_{n+1}(Z_n(x))) \end{aligned} \quad (1)$$

where  $\sigma(\cdot)$  refers to the nonlinear activation function and  $\text{CONV}_{n+1}(\cdot)$  refers to the convolutional operation whose kernel size is  $S$ , input channel number is  $C_n$  and the output channel number is  $L_n$ . As in as Eq.1, there exists the constraint  $C_{n+1} = L_n$ . In this example, there should be a layer group containing parameter matrix of  $\text{CONV}_{n+1}$  and parameter matrix of  $\text{CONV}_n$  with the relationship of  $C_{n+1} = L_n$ . The operation of grouping is necessary because the network's connectivity determines that certain layers must have consistent channel number for the network to function properly. As parameter matrices in a same group might have different redundancy rates, we analyze the group with the ACRE method to estimate the compression range which will be shared by this group.

### A.2 Sampled model and discrete model

When we fix the sample grids, there is a corresponding sampled model parameterized by  $T(\lambda)$ . We rewrite the equation of  $T(\lambda)$  based on Eq.6 in the main document as:

$$T(\lambda)_{x_{in}, x_{out}} = \sum_{ij} \lambda_{ij} u(x_{out} m_{out} - i) u(x_{in} m_{in} - j) \quad (2)$$

$T(\lambda)$  is the result of sampling  $F_W$  (see Eq.6 in the main document) on the fixed sample grids.

The discrete model is generated from the sampled model by converting the interpolated parameters  $T(\lambda)$  to parameter matrices. The main difference between the sampled model and the discrete model is that the sampled model calculates layer output based on Eq.4, Eq.5 and Eq.9 in the main document using the trapezoidal integration method, while the discrete one calculates layer output based on matrix multiplications as in Eq.3 in the main document.

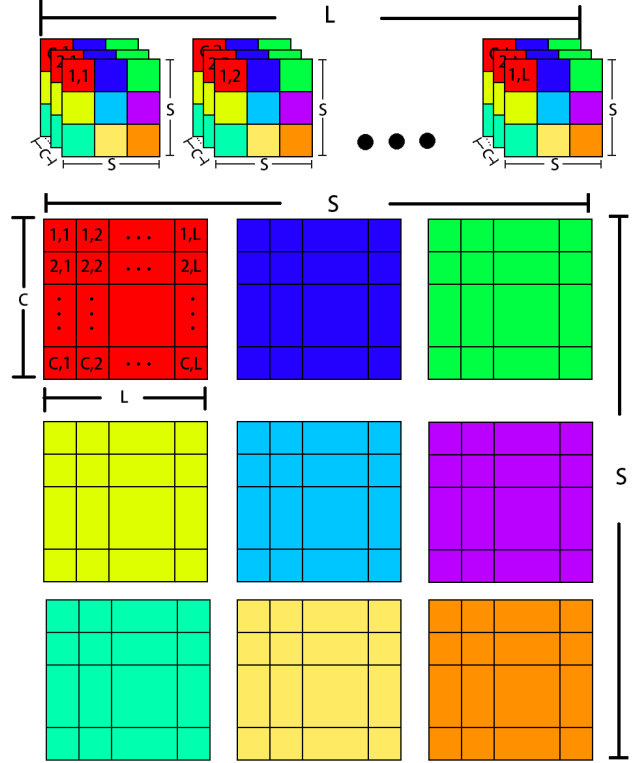


Figure 1: Example for four-dimensional tensor.

### A.3 ACRE

The parameters of a convolutional layer is a four-dimensional tensor  $y \in \mathbb{R}^{L \times C \times S \times S}$ , where  $L$  refers to the output channel length,  $C$  for the input channel number and  $S$  for the kernel size. As shown on Fig.1 for example of reshape operation, where colors represent the positions of entries rather than values, we sample from the same position of each kernel and construct  $S \times S$  matrices. The new matrix in  $\mathbb{R}^{C \times L}$  represents the transformation relationship between the input and output at a specific position. The coordinate transformation is as follows:

$$\tilde{y}(u, v, j, i) = y(i, j, u, v) \quad (3)$$

where  $(i, j, u, v)$  are the indices on the respective dimensions of  $y$ . The parameter matrices of linear transformations in FCNNs and transformers are considered as the  $1 \times 1$  particular case of  $y$ . In the ACRE method, we infer the redundancy of the convolution operation at a specific position by analyzing the redundancy of these matrices. We select the minimum redundancy among these  $S \times S$  positions as an estimate of the overall redundancy and use it to estimate the final compression rate range.

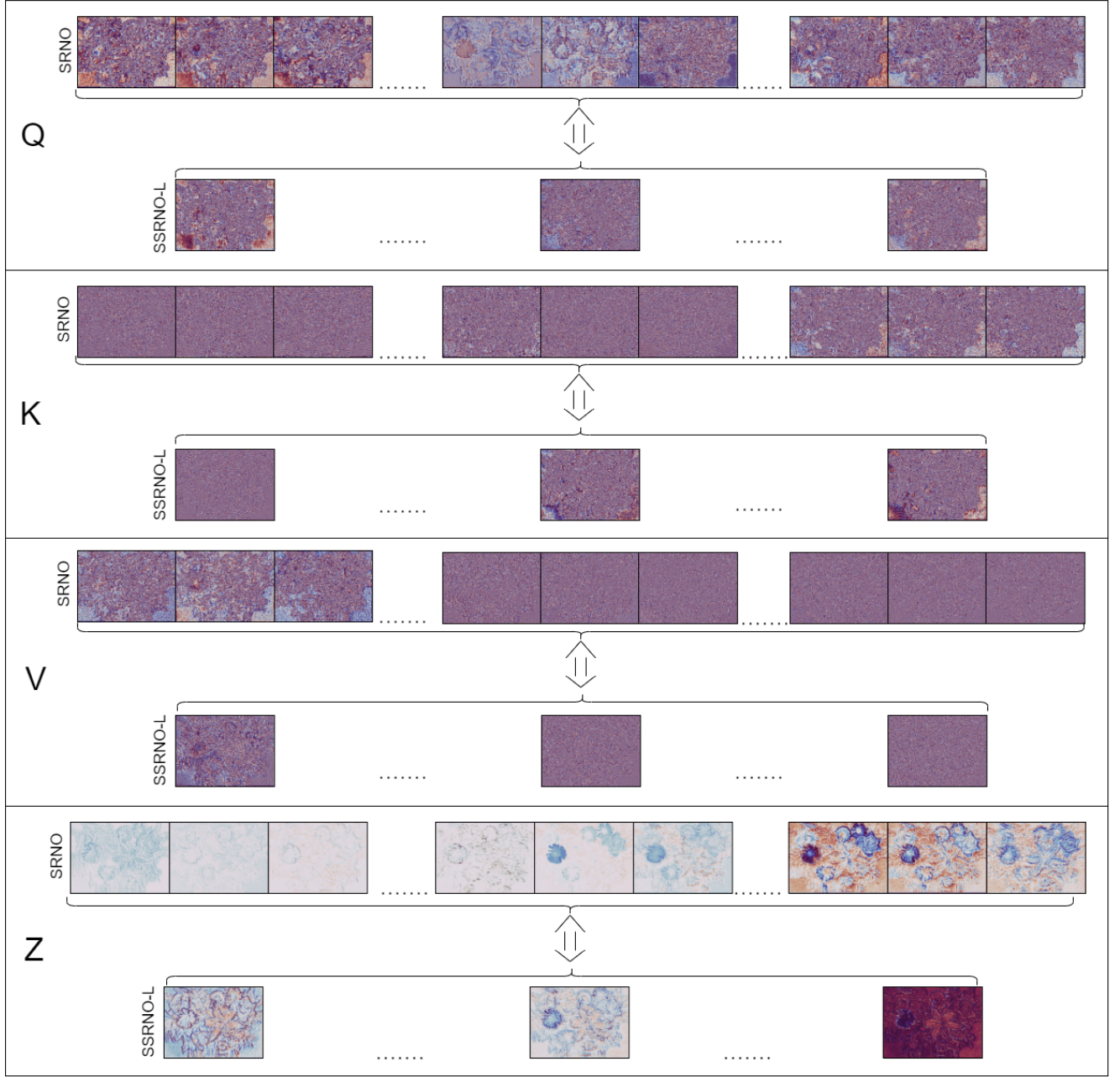


Figure 2: Basis functions visualization for SRNO[5] and SSRNO-L.  $Z$  in the figure is formulated as  $Z = Q(K^T V)$ .

## B VISUALIZATION OF BASIS FUNCTIONS

To further explain our method and its effectiveness, we present the visualization of the basis functions in SSRNO and compare it with SRNO[5]. As can be observed in Fig.2, the basis functions in SRNO are redundant. The basis functions in SSRNO are not only more diverse but also contain richer textures than in SRNO. The spaces spanned by SSRNO and SRNO are equivalent and this is one of the fundamental reasons why we can achieve good compression results

in the Galerkin attention layers while maintaining the performance.

## C ADDITIONAL VISUALIZATION RESULTS

To better showcase the performance of SSRNO, in this section, we present more visualization results on image reconstruction in Fig.3. We also provide the results by SRNO[5], LIIF[1] and LTE[3] for comparisons.

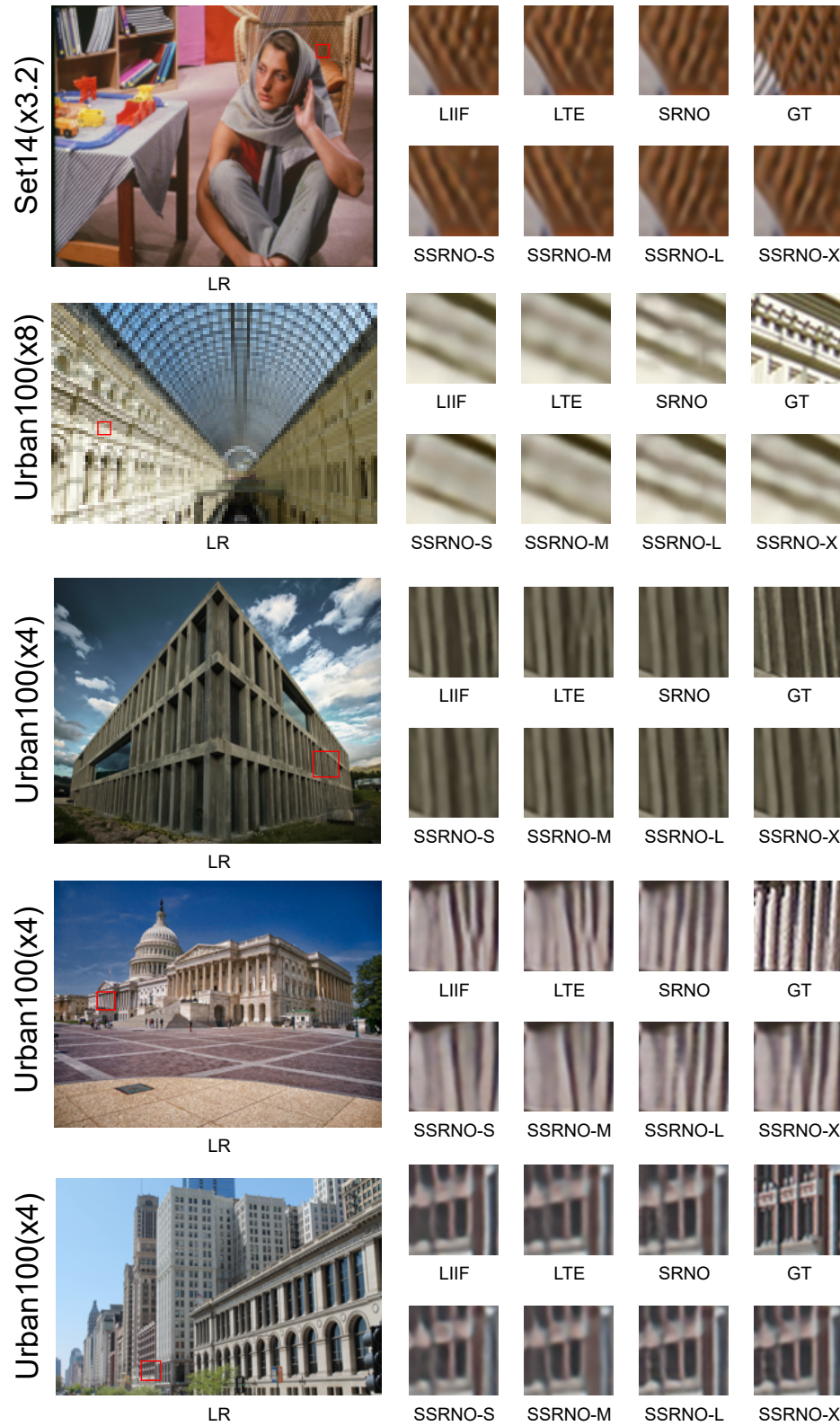


Figure 3: Visual comparison on more images[2, 4, 6]. The boxes in the LR image represents the area that we focus on in different model results. All methods are trained with the scales of  $\times 1\text{-}\times 4$  and use EDSR as their encoder.

REFERENCES

[1] Yinbo Chen, Sifei Liu, and Xiaolong Wang. 2021. Learning Continuous Image Representation With Local Implicit Image Function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8628–8638.

[2] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. 2015. Single Image Super-Resolution From Transformed Self-Exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[3] Jaewon Lee and Kyong Hwan Jin. 2022. Local Texture Estimator for Implicit Representation Function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1929–1938.

[4] D. Martin, C. Fowlkes, D. Tal, and J. Malik. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and

measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, Vol. 2. 416–423 vol.2. <https://doi.org/10.1109/ICCV.2001.937655>

[5] Min Wei and Xuesong Zhang. 2023. Super-Resolution Neural Operator. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)*, 18247–18256. <https://api.semanticscholar.org/CorpusID:257365077>

[6] Roman Zeyde, Michael Elad, and Matan Protter. 2012. On Single Image Scale-Up Using Sparse-Representations. In *Curves and Surfaces*, Jean-Daniel Boissonnat, Patrick Chenin, Albert Cohen, Christian Gout, Tom Lyche, Marie-Laurence Mazure, and Larry Schumaker (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 711–730.

349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406

407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464