Alma Mater Studiorum University of Bologna

Artificial Intelligence - Machine Learning for Computer Vision

Project work

Alessandro Dicosola [Matr. 935563]

Task

SR is an Ill-posed problem: trying to create a mapping between LR to HR for reconstructing SR images
Keywords: super-sampling, super-resolution.

Related works

SR through internal and external database[2, Releated works]

Exploit similarity between textures in images therefore create pairs of LR-HR patches for learning a low dimensional space were apply k-NN, non linear regressor, random forest, ... , manifold embedding[1, 2004]:

- LR patches embedding is the concatenation of the first and second order derivative of the luminance at each pixel on the patch in the YIQ domain.
- HR patches embedding is the concatenation of all luminance at each pixel
- Given a LR patch find the k-NN patches on the training set

$$N_q$$

- Compute the reconstruction weights minimizing the reconstruction error:

$$argmin_W ||x_t^q - \sum_{x_s^p \in N_q} w_{qp} x_s^p||^2$$

- Reconstruct the HR patch:

$$y_t^q = \sum_{x_s^p \in N_q} w_{qp} y_s^p$$

Figure 6. 3X magnification of the bird image: (a) input low-resolution image; (b) true high-resolution image; (c) median filtering; (d) cubic spline interpolation; (e) our method with training images shown in Figures 2(c) and (d).

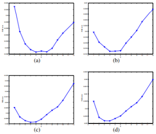


Figure 8. RMS error between the super-resolution image generated and the ground-truth image as a function of the number of nearest neighbors used: (a) head image; (b) lizard image; (c) bird image; (d) plant image.

Related works

CNN [10, Releated works] [11]

- SRCNN [3] : CNN using only three convolutions for extracting information, process them and reconstruct the SR image.
- VDSR : VGG-net that learns a residual instead of the direct mapping LR-HR in order to focus the training on high frequency information.
- SRDenseNet, RDN, DBDN[4]: Combine residual connection (locally and globally) with dense connection
- RCAN[5], CSFM[6] : Use channel-wise and spatial attention for focusing the training on important features or region in order to ease the convergence and improve the results.
- LapSRN[2], MS-LapSRN[7] : multi scale network using a *Laplacian Pyramid Framework* [8]
- MetaSR[9] : use a meta-upscaling module for learning LR-HR mapping using any scale.

ASDN - Arbitrary-Scale Deep Network

- Find an LR-HR mapping with any scale (integer or real)
- Use Laplacian Frequency Representation [8] for reconstructing SR image interpolating two nearest level in the LFR.
- The range scales was found to be optimal between 1 and 2 using 11 levels in the LFR.
- For scale greater than 2 the network is *deployed recursively*.

ASDN - Arbitrary-Scale Deep Network

Laplacian Frequency Representation

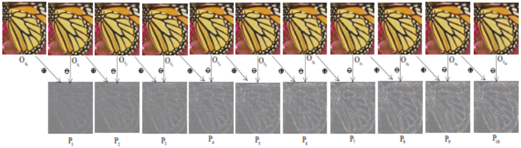


Fig. 2: The Laplacian Frequency Representation has 11 Laplacian pyramid levels ($O_{r_0}, ..., O_{r_{10}}$), with 10 phases in the scale range of $(1, 2]$ ($P_1, ..., P_{10}$). Each phase represents the difference between the successive Laplacian pyramid levels.

- Learning any decimal scale is not possible: infinite decimal scales.
- Allow to have reduced dataset.
- Reconstruct the SR image interpolating the two nearest level in the LFR.
- Each level learn the HR representation at particular scale in the range.
- Compute the *phase* as the difference between two consecutive levels in order to acquire **high frequency information**:

$$P_i = O_{r_{i-1}} - O_{r_i}$$

ASDN - Arbitrary-Scale Deep Network

Laplacian Frequency Representation

- The scale for each level is:

$$r_l = \frac{l}{L-1} + 1$$

therefore the level can be indexed as

$$l = (r_l - 1) * (L - 1)$$

$$i = \lceil (r - 1) * (L - 1) \rceil$$

- Define the weight, proportional to the distance between the scales, as:

$$w_r = (L - 1) * (r_i - r)$$

- The SR image is reconstructed as:

$$O_r = O_{r_i} + w_r * P_i$$

ASDN - Arbitrary-Scale Deep Network

Example

$$r = 1.27$$

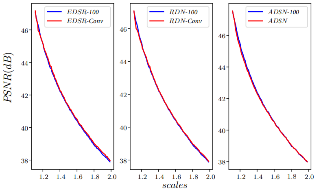
$$i = \lceil 10 * (1.27 - 1) \rceil = \lceil 2.7 \rceil = 3$$

$$w_r = 10 * (1.3 - 1.27) = 0.3$$

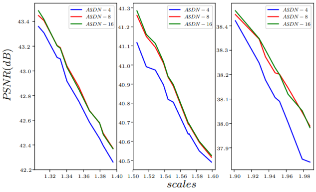
$$O_r = O_{r_3} + 0.3 * P_3$$

ASDN - Arbitrary-Scale Deep Network

Study of Laplacian Frequency Representation



(a) Study of Effectiveness



(b) Study of Density

Fig. 8: Study of Laplacian Frequency Representation

- ESDR-100, RDN-100, ASDN-100: upsampling networks trained and tested on 100 scales between (1,2] on Set5.
- ESDR-Conv, RDN-Conv, ADSN: network with 11 IRBs.
- ASDN-4,8,16: networks with 5,9,17 levels in the LFR with a scales range between (1,2]
- A denser LFR is better but saturate beyond 10 phases (11 levels).

ASDN - Arbitrary-Scale Deep Network

Recursive deployment

- Learning the mapping for any scale is too complex.
- For scales greater than the maximum in the scale range apply recursively the network.
- A large scale R can be expressed as an integer N power of decimal ratio r:

$$R = r^N \Rightarrow N = \log_r R$$

- Was found that using the greatest r at the beginning lead to better performance
- Since the range is fixed between (1,2] then

$$\lceil N = \log_2 R \rceil$$

and therefore

$$r_i = \begin{cases} 2 & \text{i } \leq \text{N - 1} \\ \frac{R}{2^{N-1}} & \text{i} = \text{N} \end{cases}$$

ASDN - Arbitrary-Scale Deep Network

Study of recursive deployment - Effectivness

Methods	Direct deployment			Recursive deployment		
	$\times 2$	$\times 3$	$\times 4$	$\times 2$	$\times 3$	$\times 4$
VDSR-Conv	37.57	33.77	31.56	36.86	33.70	31.50
EDSR-Conv	38.04	34.45	32.29	37.18	34.32	32.26
RDN-Conv	38.05	34.46	32.31	37.27	34.38	32.23
Ours	38.12	34.52	32.28	37.35	34.43	32.27

Table 3: PSNR of the recursive deployment and direct deployment on SR for $\times 2, \times 3, \times 4$

- *recursive deployment*: VDSR-Conv, EDSR-Conv, RDN-Conv, ASDN trained on 11 decimal scales in (1,2] and tested with HR images upscaled twice

$$\sqrt{2}, \sqrt{3}, \sqrt{4}$$

in order to generate X2, X3, X4 HR images (therefore 2 recursions).

- *direct deployment*: VDSR-Conv, EDSR-Conv, RDN-Conv, ASDN trained on images upsampled directly X2,X3,X4.
- **Recursive deployment** lead to worst performance but the difference goes down with the scale.

ASDN - Arbitrary-Scale Deep Network

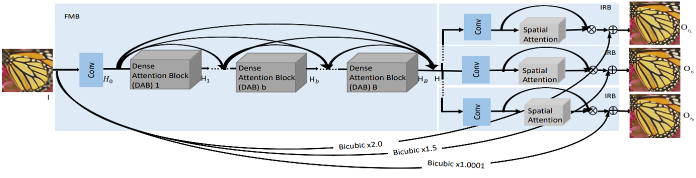
Study of recursive deployment - Optimal N and r

Scale(R)	Recursion(N)	UpscaleRatio(r)	PSNR
3×	2	1.732, 1.732	34.43
		1.500, 2.000	34.19
		2.000, 1.500	**34.48**
	3	1.442, 1.442, 1.442	33.18
4×	2	2.000, 2.000	**32.27**
	3	1.587, 1.587, 1.587	31.96
		1.800, 1.800, 1.234	32.16
		2.000, 1.800, 1.100	32.24

- Different combination of N and r.
- Use the greatest r in the scales range at the beginning lead to better performance.

ASDN - Arbitrary-Scale Deep Network

Architecture



(a) Network structure of Any-scale Deep SR Network (ASDN)

Dense Attention Block (DAB) b

(b) Illustration of Dense Attention Block (DAB)

Spatial Attention(SA)

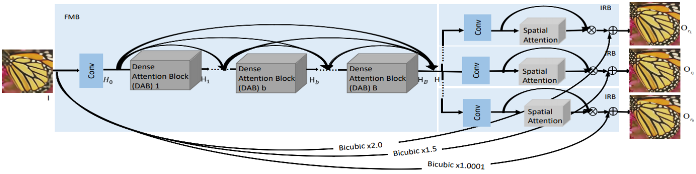
Channel Attention(CA)

(c) Illustration of Spatial Attention module (SA) and Channel Attention (CA)

- Bi-dense connections (DAB with IDB) for learning efficiently features (since DenseNet, SRDenseNet and RDN doesn't allow previous features to be used into next blocks).
- IDB contains convolutions whose activations are processed with ReLu (without BN) and densely connected.
- CA from RCAN in order to focus on most important features.
- SA from CSFM in order to focus on particular and important textures at the reconstruction.
- Two IRBs are activated based on the scale used for training or testing and the the final image is reconstructed using the formula seen before.
- Global skip connection for including low frequency information from the input and allow the network to learn high frequency information only.

ASDN - Arbitrary-Scale Deep Network

Architecture



(a) Network structure of Any-scale Deep SR Network (ASDN)

(b) Illustration of Dense Attention Block (DAB)

(c) Illustration of Spatial Attention module (SA) and Channel Attention (CA)

- All convolutions have 64 filters and 3x3 kernel size with 0-padding and stride equals to 1 but the ones in IRB (1x1 kernel and 3 filters), CA (1x1 kernel which reduce-expand the channels), SA (1x1 kernel which expand-reduce the channels).
- Number of DABs: 16.
- Number of convolutions inside IDBs: 8.

Extra

Laplacian pyramid[8]

- Generate images with high frequency information only subtracting filtered images with itself non filtered: similar to apply a laplacian operator on the image.
- Remove correlation between pixels.
- Those images can be used for reconstructing the original image.
- Hand crafted kernel for filtering the image in order to cutoff low frequency (low-pass filtering) information:
  - Separable
  - Symmetric
  - Normalized:
    $$a + 2b + 2c = 1$$
  - Equal contribution:
    $$a + 2c = 2b$$
  - Kernel =

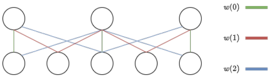


Figure: Equal contribution.

- All nodes in the lower level must have the same contribution to nodes in the upper level.
- The central node contributes with a + 2c
- The middle ones contribute with 2b
- The outer ones with a + c
- So a+2c=2b

Extra

Laplacian pyramid[8]

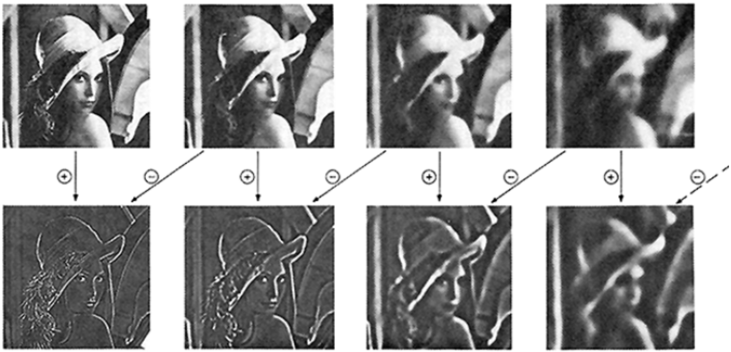


Fig 5. First four levels of the Gaussian and Laplacian pyramid. Gaussian images, upper row, were obtainedby expanding pyramid arrays (Fig. 4) through Gaussian interpolation. Each level of the Laplacian pyramid is the difference between the corresponding and next higher levels of the Gaussian pyramid.

Hong Chang, Dit-Yan Yeung, and Yimin Xiong, "Super-resolution through neighbor embedding," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, pp. I–I, 2004.

W. Lai, J. Huang, N. Ahuja, and M. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," *CoRR*, vol. abs/1704.03915, 2017.

C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *CoRR*, vol. abs/1501.00092, 2015.

Y. Wang, J. Shen, and J. Zhang, "Deep bi-dense networks for image super-resolution," *CoRR*, vol. abs/1810.04873, 2018.

Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," *CoRR*, vol. abs/1807.02758, 2018.

- Y. Hu, J. Li, Y. Huang, and X. Gao, “Channel-wise and spatial feature modulation network for single image super-resolution,” *CoRR*, vol. abs/1809.11130, 2018.
- W. Lai, J. Huang, N. Ahuja, and M. Yang, “Fast and accurate image super-resolution with deep laplacian pyramid networks,” *CoRR*, vol. abs/1710.01992, 2017.
- P. Burt and E. Adelson, “The laplacian pyramid as a compact image code,” *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.
- X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, “Meta-sr: A magnification-arbitrary network for super-resolution,” *CoRR*, vol. abs/1903.00875, 2019.
- J. Shen, Y. Wang, and J. Zhang, “Asdn: A deep convolutional network for arbitrary scale image super-resolution,” *ArXiv*, vol. abs/2010.02414, 2020.
- S. Anwar, S. Khan, and N. Barnes, “A deep journey into super-resolution: A survey,” *CoRR*, vol. abs/1904.07523, 2019.