Video Super-Resolution Based on Spatial-Temporal Recurrent Residual Networks Supplementary Material

Wenhan Yang[a], Jiashi Feng[b], Guosen Xie[c], Jiaying Liu[a], Zongming Guo[a], and Shuicheng Yan[d]

[a] *Institute of Computer Science and Technology, Peking University, Beijing, P.R.China*

[b] *Department of Electrical and Computer Engineering, National University of Singapore*

[c] *NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, P.R.China*

[d] *Artificial Intelligence Institute, Qihoo 360 Technology Company, Ltd., Beijing, P.R.China*

Abstract

This supplementary material provides more empirical analysis and discussions on STR-ResNet for video SR.

Contents

1. More Analysis and Discussions

1.1. Ablation Analysis

We here perform ablation studies to investigate the individual contribution of each component in our model to the final performance. We use following notations to represent each version of our proposed STR-ResNet as shown in Fig. 1,

- *BRCN*. A three layer recurrent convolution network in [1] which is used as our baseline.

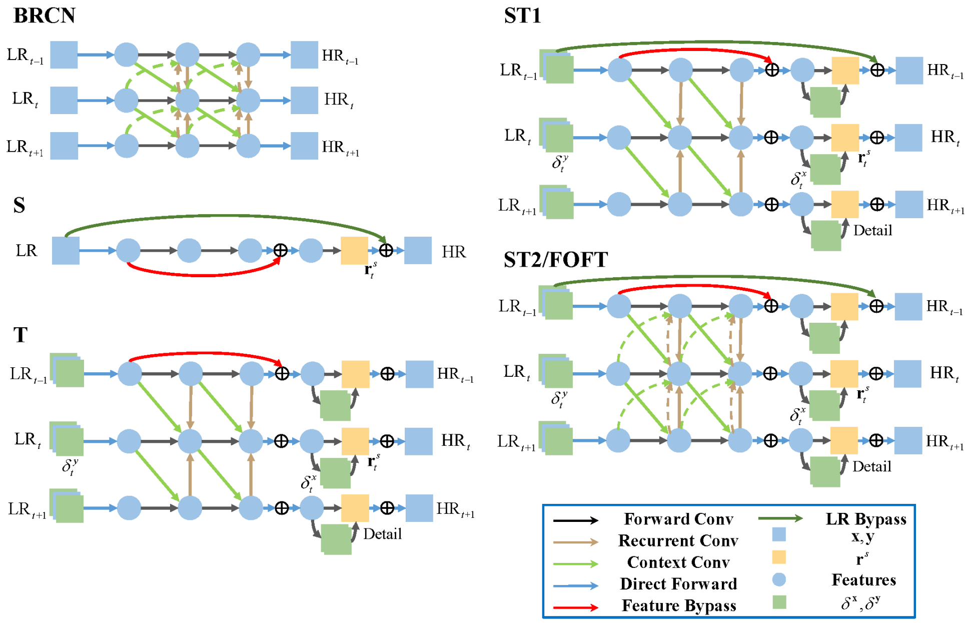

Figure 1: The network architectures of the compared versions in ablation analysis.

- *S*. A six layer recurrent convolution network with only spatial residual learning, modeled by bypass connections.
- *T*. BRCN with only temporal residual learning, modeled by LR difference frame inputs. Without the aid of spatial residual and bypass connection, we could only set the depth of the network as 3.
- *ST1*. S with both spatial and temporal residuals but in only one direction.
- *ST2*. S with both spatial and temporal residuals in two directions.
- *FOFT*. A finetuned version of ST2, with the centric exponential decayed weighted loss, as shown in Table 1 of the main body.

Their performances and parameter numbers are presented in Table 1. Note that, the parameter numbers of all methods are presented on the basis of that of *S*. The versions BRCN, T, ST1 and ST2 can output 9 HR frames at a time, thus their parameter numbers are divided by 9. It is observed that, adding spatial and temporal residues individually contributes little to the performance of the network. It shows that without a joint consideration of both kinds of residues, adding a single term into the model only brings limited performance gain. Simply increasing the depth of the network only leads to slight performance gain of

0.09dB (S). Also, without a deeper structure, introducing inter-frame information does not significantly boost the performance (T) – the average performance gain is only 0.02dB. Modeling both spatial and temporal residuals (ST1) overcomes these deficiencies, leading to significantly boosted performance (0.42dB). Adding two-way connections (ST2) also improves the network performance than the version with one-way connection, with an average performance gain of 0.12dB. The finetuning (FOFT) with the centric exponential decayed weighted losses also benefits SR estimation, with an average performance gain of 0.02dB .

Table 1: The ablative analysis results for each component of STR-ResNet.

Versions	BRCN	T	S	ST1	ST2	FOFT
#Para	5	3	1	3	5	45
Tractor	33.23	33.32	33.32	33.74	33.84	**33.85**
Sunflower	39.28	39.30	39.35	39.60	39.96	**40.02**
Blue sky	31.40	31.50	31.50	32.13	32.24	**32.23**
Station	35.20	35.23	35.28	35.61	35.61	**35.63**
Pedestrian	34.95	34.88	35.01	35.18	35.18	**35.22**
rush_hour	39.86	39.81	39.96	40.15	40.28	**40.30**
Average	35.65	35.67	35.74	36.07	36.19	**36.21**

Table 2: PSNR results among different methods for video SR (scaling factor: 4).

Versions	FOFT	ST2	S-P9	S-128
#Para	5	45	45	4
Tractor	33.84	33.85	33.41	33.38
Sunflower	39.96	40.02	39.52	39.43
Blue Sky	32.24	32.23	31.75	31.57
Station	35.61	35.63	35.43	35.34
Pedestrian	35.18	35.22	35.08	35.03
Rush Hour	40.28	40.30	40.12	40.11
Average	36.19	36.21	35.89	35.81

1.2. Comparing with Larger Networks with Single Frame Input

To demonstrate the source of our gains, we further compare ST2 and FOFT with another two versions of our methods: S-P9 and S-128. S-P9 owns the same network structure to ST2, but its 9 sub-networks take the center LR frame as their inputs. S-128 has only one sub-network, and its convolutional layers in the middle have 128 channels. The PSNR results among these four methods are presented in Table 2. It is clearly demonstrated that, increasing parameters can boost the performance. However, the gains are limited, compared with those brought by using adjacent frames via the joint spatial-temporal structure.

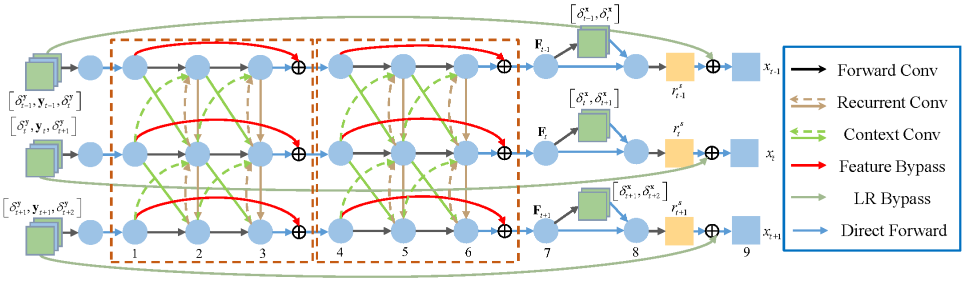

Figure 2: The architecture of STR-ResNet that exploits the adjacent frames in two temporal directions.

1.3. Video SR without Future Frames

We first need to mention that, the network structure in Fig. 5 of the main body is not the exact final network structure of STR-ResNet. The temporally backward convolutions are omitted for a clearer illustration. The final network structure of STR-ResNet that exploits the adjacent frames in two temporal directions are presented as shown in Fig. 2.

Without using future frames, STR-ResNet can be simplified via three steps as shown in Fig. 3:

1. We remove the convolution connections from the future frames and only enable the information flow from the past frames to the current frame;
2. The network takes only LR frames and the temporal residue between the past LR frame and the current LR frame as its input;
3. In the penultimate layer of the network, only the temporal residue between the past HR frame and the current HR frame is predicted.

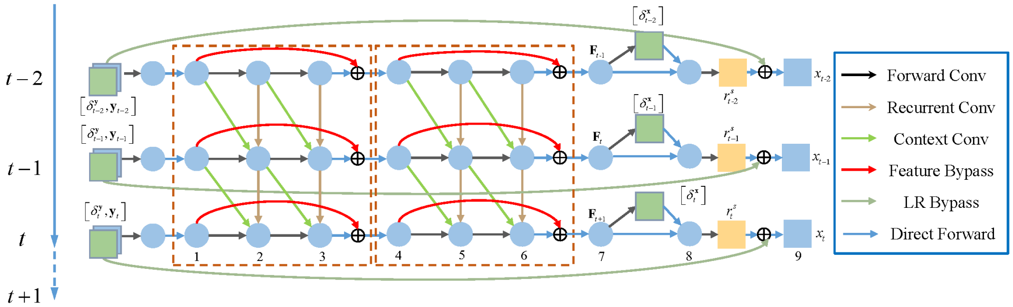

Figure 3: The architecture of STR-ResNet that exploits the information of adjacent frames in two temporal directions.

Therefore, STR-ResNet can predict the current HR frame solely relying on the past frames and the processing delay will be reduced.

1.4. Handling Color Videos

Our method is flexible to handle color video super-resolution. We achieve that in two ways : 1) using single-channel input/output STR-ResNets to process each channel separately; 2) using a three-channel input/output STR-ResNet to process three channels jointly. The results are presented in Table 3. B-STR-ResNet denotes the version where chromatic channels are processed by Bicubic interpolator. C-STR-ResNet denotes the version where three channels of YCbCr are processed separately by single-channel input/output STR-ResNets, respectively. JC-STR-ResNet denotes the version where three channels of YCbCr are processed jointly by a three-channel input/output STR-ResNet. Comparing B-STR-ResNet and C-STR-ResNet, it is clearly demonstrated that, replacing Bicubic interpolator by single-channel input/output STR-ResNets in processing chromatic channels slightly improves the reconstruction quality. A joint processing for luminance and chrominance in JC-STR-ResNet leads to no significant gain compared with C-STR-ResNet. The joint training of JC-STR-ResNet in RGB space leads to a large performance drop.

Table 3: PSNR results in RGB color space among different methods for video SR (scaling factor: 4).

Video	B-STR-ResNet	C-STR-ResNet	JC-STR-ResNet	JC-STR-ResNet
Color Space	YCbCr	YCbCr	YCbCr	RGB
Tractor	30.63	30.85	30.82	30.21
Sunflower	34.01	34.24	34.29	33.76
Blue Sky	29.86	30.11	30.12	29.45
Station	32.40	32.56	32.52	32.01
Pedestrian	33.11	33.24	33.29	32.80
Rush Hour	36.92	37.21	37.23	36.30
Average	32.82	33.04	33.05	32.42

1.5. Benefits of Residual CNNs

The usage of residual CNNs for single frame image processing tasks has been proved effective in many research topics, including image super-resolution [2], image denoising [3], and single image rain removal [4]. For the reasons of its outstanding performance, we give our understandings as follows,

- Network training is a non-convex optimization problem. With the same input and expected output, the performance of a network in practice is decided by many factors, including network structure, optimization methods, training data *et al.*

- Residual CNNs faces fewer chances to stop at local minima. Comparing with CNNs that model full images, Residual CNNs only need to fit the residual signal with lower energy, and the information through the network is reduced. Thus, the network training converges faster and to a better solution, as the empirical evaluations illustrated in [5, 2].

- Better decorrelation. Usually, the HR image $\mathbf{x}$ is highly correlated to the LR one $\mathbf{y}$. Especially at pixel level, $\mathbf{x}(i, j)$ is highly correlated to not only $\mathbf{y}(i, j)$ but also $\mathbf{y}(k, l)$ where $(k, l) \in \epsilon\,(i, j)$, the neighbors of (i, j). Thus, when regressing $\mathbf{x}(i, j)$, many pixels in the same region of $\mathbf{y}$ contribute to it. This is usually harmful for the network training [6]. Residual CNN gets rid of this issue by only learning to restore the residual signals. In fact, similar ideas have been proved effective in many conventional methods, such as ScSR [7] and A+ [8]. In these methods, the image patches are also preprocessed to remove the redundant low frequency signals for better modeling high-frequency details.

- More structural correspondences. Conventional and residual CNNs can be regarded as filters. It is usually beneficial for a filter to work on a domain where more structural correspondences are provided. We have discussed this point more clearly as shown in Fig. 2 and in Section 3 of the main body.

Table 4: The effect of number of time/recurrence steps of STR-ResNet on video SR performance and computational cost.

Video	3	5	7	9
Tractor	33.63	33.76	33.78	**33.85**
Sunflower	39.53	39.64	39.76	**40.02**
Blue Sky	31.93	32.11	32.15	**32.23**
Station	35.40	35.54	35.59	**35.63**
Pedestrian	35.10	35.11	35.15	**35.22**
Rush Hour	40.25	40.27	40.27	**40.30**
Ave. PSNR (Db)	35.97	36.07	36.12	**36.21**
Ave. Time (s)	47.1472	71.9190	108.4440	124.9450

1.6. Analysis on Time/Recurrence Step Number

We investigate how the number of time or recurrence steps in the STR-ResNet influences the SR performance. We vary the number of steps from 3 to 9 and evaluate the performance of corresponding models. Table 4 shows that, increasing recurrence steps to model adjacent frames consistently improves the reconstruction performance which also introduces reasonably higher computational cost as expected. The step number of 9 gives the best performance and the computational cost is still acceptable.

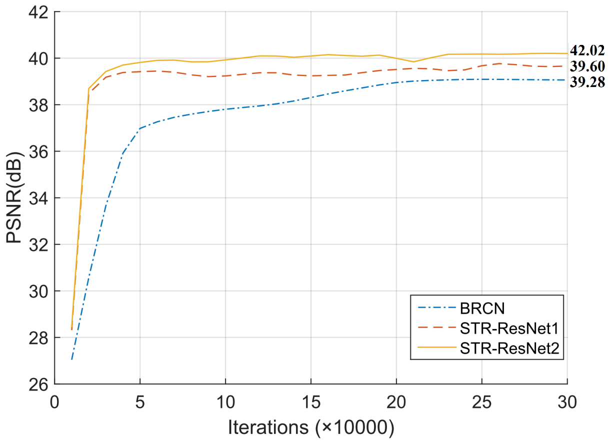

Figure 4: The performance comparison of BRCN and two versions of the proposed STR-ResNet on the validation set during training.

1.7. Validation Performance in Training Process

To investigate the training behavior of STR-ResNet, we use the sequence *blue sky* as the validation set and present its validation performance during the training for BRCN and the proposed STR-ResNet, including both one-direction and two-direction versions, as shown in Fig. 4. It shows that, adding spatial and temporal residuals speeds up the convergence of STR-ResNet. STR-ResNet converges faster than BRCN and achieves better SR performance. It is very interesting to see that, PSNRs of three methods increase very fast in the first 50000 iterations (first 6 epochs). STR-ResNet1 and STR-ResNet2 achieve almost the same evaluation performance in the first 20000 iterations (first 3 epochs). After that, STR-ResNet2 achieves better performance benefiting from receiving information in two directions.

1.8. Situations of Temporal Residues Being Useful

To observe the performance of SR methods with / without temporal residues in each situation, we design a metric to visualize their performance comparison. We first calculate the Mean Square Errors (MSE) between the patches of the SR results with / without temporal residues and the corresponding patches of the HR image. Then, we use the patch MSE ratio to signify the regions where adding temporal residues leads to a performance gain or not. The results are presented in Fig. 5.

The regions where adding temporal residues leads to a performance gain are denoted in blue and the regions where adding temporal residues leads to a performance loss are denoted in red. It is clearly shown that, in texture abundant regions of *Tractor*, *Blue Sky* and *Rush Hour* sequences, adding temporal

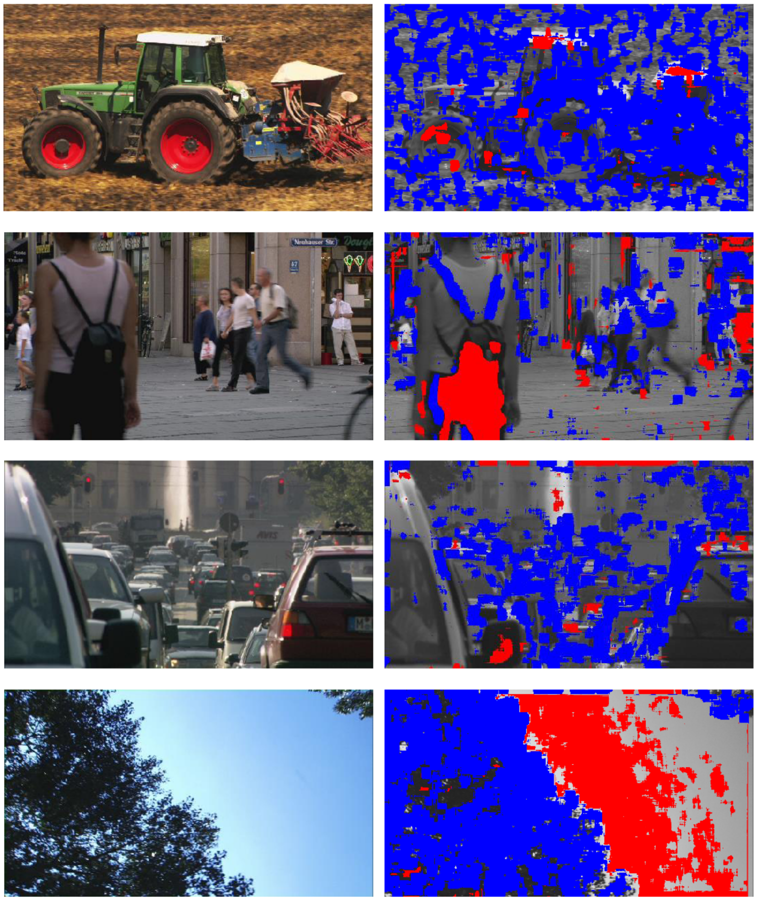

Figure 5: Analysis on the situations of adding temporal residues being useful. The regions where adding temporal residues leads to a performance gain are denoted in blue and the regions where adding temporal residues leads to a performance loss are denoted in red.

residues has an overwhelming advantage. Comparatively, in the smooth regions, *i.e.* the bag in *Pedestrian* and the sky in *Blue Sky*, the version without temporal residues has an advantage. In all, as shown in Tables 2 and 3 of the main body, adding temporal residues provides overall performance gains in PSNR and SSIM.

References

[1] Y. Huang, W. Wang, L. Wang, Bidirectional recurrent convolutional networks for multi-frame super-resolution, in: Proc. Annual Conference on Neural Information Processing Systems, 2015, pp. 235–243.

[2] J. Kim, J. K. Lee, K. M. Lee, Deeply-recursive convolutional network for image super-resolution, in: Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, 2016, pp. 1637–1645. doi:10.1109/CVPR.2016.181.

[3] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising, IEEE Transactions on Image Processing 26 (7) (2017) 3142–3155. doi:10.1109/TIP.2017.2662206.

[4] J. F. J. L. Z. G. Wenhan Yang, Robby T. Tan, S. Yan, Joint rain detection and removal from a single image, Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition.

[5] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[6] M. Cogswell, F. Ahmed, R. B. Girshick, L. Zitnick, D. Batra, Reducing overfitting in deep networks by decorrelating representations, ICLR.

[7] J. C. Yang, J. Wright, T. S. Huang, Y. Ma, Image super-resolution via sparse representation, IEEE Transactions on Image Processing 19 (11) (2010) 2861–2873.

[8] R. Timofte, V. De Smet, L. Van Gool, A+: Adjusted anchored neighborhood regression for fast super-resolution, in: Proc. IEEE Asia Conf. Computer Vision, 2014.