

End-to-End Learning of Video Super-Resolution with Motion Compensation

Osama Makansi, Eddy Ilg, and Thomas Brox

Department of Computer Science, University of Freiburg

Abstract. Learning approaches have shown great success in the task of super-resolving an image given a low resolution input. Video super-resolution aims for exploiting additionally the information from multiple images. Typically, the images are related via optical flow and consecutive image warping. In this paper, we provide an end-to-end video super-resolution network that, in contrast to previous works, includes the estimation of optical flow in the overall network architecture. We analyze the usage of optical flow for video super-resolution and find that common off-the-shelf image warping does not allow video super-resolution to benefit much from optical flow. We rather propose an operation for motion compensation that performs warping from low to high resolution directly. We show that with this network configuration, video super-resolution can benefit from optical flow and we obtain state-of-the-art results on the popular test sets. We also show that the processing of whole images rather than independent patches is responsible for a large increase in accuracy.

1 Introduction

The task of providing a good estimation of a high-resolution (HR) image from low-resolution (LR) input with minimum upsampling effects, such as ringing, noise, and blurring has been studied extensively [10,4,24,25]. In recent years, deep learning approaches have led to a significant increase in performance on the task of image super-resolution [7,13,14,15]. Potentially, multiple frames of a video provide extra information that allows even higher quality up-sampling than just a single frame. However, the task of simultaneously super-resolving multiple frames is inherently harder and thus has not been investigated as extensively. The key difficulty from a learning perspective is to relate the structures from multiple frames in order to assemble their information to a new image.

Kappeler et al. [12] were the first who proposed a convolutional network (CNN) for video super-resolution. They excluded the frame registration from the learning problem and rather applied motion compensation (warping) of the involved frames using precomputed optical flow. Thus, only a small part of the video super-resolution task was learned by the network, whereas large parts of the problem rely on classical techniques.

In this work, we provide for the first time an end-to-end network for video super-resolution that combines motion compensation and super-resolution into a single network with fast processing time. To this end, we make use of the FlowNet2-SD for optical flow estimation [11], integrate it into the approach by Kappeler et al. [12], and train the joint network end-to-end. The integration requires changing the patch-based training [7,12] to an image-based training and we show that this has a positive effect. We analyze the resulting approach and the one from Kappeler et al. [12] on single, multiple, and multiple motion-compensated frames in order to quantify the effect of using multiple frames and the effect of motion estimation. The evaluation reveals that with the original approach from Kappeler et al. both effects are surprisingly small. Contrary, when switching to image-based training we see an improvement if using motion compensated frames and we obtain the best results with the FlowNet2-SD motion compensation.

The approach of Kappeler et al. [12] follows the common practice of first upsampling and then warping images. Both operations involve an interpolation by which high-frequency image information is lost. To avoid this effect, we then implement a motion compensation operation to directly perform upsampling and warping in a single step. We compare to the closely related work of Tao et al. [23] and also perform experiments with their network architecture. Finally, we show that with this configuration, CNNs for video super-resolution clearly benefit from optical flow. We obtain state-of-the-art results.

2 Related work

2.1 Image super-resolution

The pioneering work in super-resolving a LR image dates back to Freeman et al. [10], who used a database of LR/HR patch examples and nearest neighbor search to perform restoration of a HR image. Chang et al. [4] replaced the nearest neighbor search by a manifold embedding, while Yang et al. built upon sparse coding [24,25]. Dong et al. [7] proposed a convolutional neural network (SRCNN) for image super-resolution. They introduced an architecture consisting of the three steps patch encoding, non-linear mapping, and reconstruction, and showed that CNNs outperform previous methods. In Dong et al. [5], the three-layer network was replaced by a convolutional encoder-decoder network with improved speed and accuracy. Shi et al. [19] showed that performance can be increased by computing features in the lower resolution space. Recent work has extended SRCNN to deeper [13] and recursive [14] architectures. Ledig et al. [15] employed generative adversarial networks.

2.2 Video super-resolution

Performing super-resolution from multiple frames is a much harder task due to the additional alignment problem. Many approaches impose restrictions, such

as the presence of HR keyframes [20] or affine motion [2]. Only few general approaches exist. Liu et al. [16] provided the most extensive approach by using a Bayesian framework to estimate motion, camera blur kernel, noise level, and HR frames jointly. Ma et al. [17] extended this work to incorporate motion blur. Takeda et al. [22] followed an alternative approach by considering the video as a 3D spatio-temporal volume and by applying multidimensional kernel regression.

A first learning approach to the problem was presented by Cheng et al. [6], who used block matching to find corresponding patches and applied a multi-layer perceptron to map the LR spatio-temporal patch volumes to HR pixels. Kappeler et al. [12] proposed a basic CNN approach for video-super-resolution by extending SRCNN to multiple frames. Given the LR input frames and optical flow (obtained with the method from [9]), they bicubically upsample and warp distant time frames to the current one and then apply a slightly modified SRCNN architecture (called VSR) on this stack. The motion estimation and motion compensation are provided externally and are not part of the training procedure.

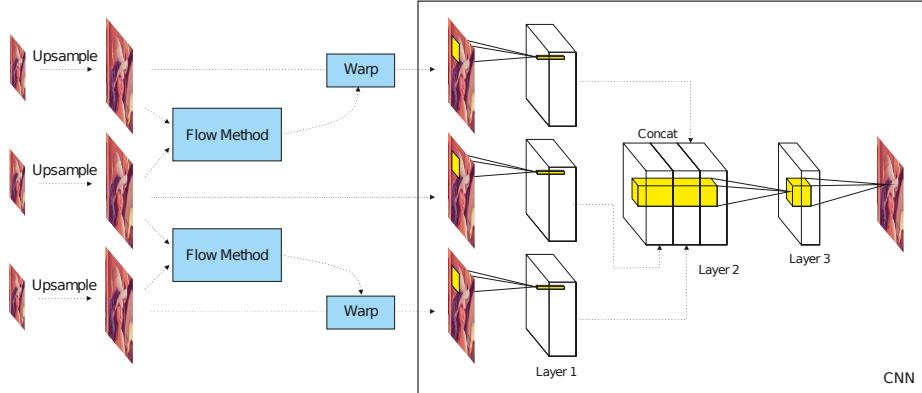
Caballero et al. [3] proposed a spatio-temporal network with 3D convolutions and slow fusion to perform video super-resolution. They employ a multi-scale spatial transformer module for motion compensation, which they train jointly with the 3D network. Very recently, Tao et al. [23] used the same motion compensation transformer module. Instead of a 3D network, they proposed a recurrent network with an LSTM unit to process multiple frames. Their work introduces an operation they call SubPixel Motion Compensation (SPMC), which performs forward warping and upsampling jointly. This is strongly related to the operation we propose here, though we use backward warping combined with a confidence instead of forward warping. Moreover, we use a simple feed-forward network instead of a recurrent network with an LSTM unit, which is advantageous for training.

2.3 Motion estimation

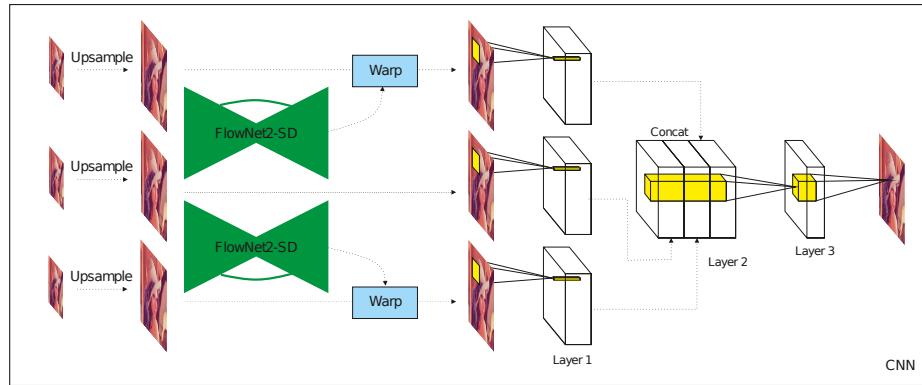
Motion estimation is a longstanding research topic in computer vision, and a survey is given in [21]. In this work, we aim to perform video super-resolution with a CNN-only approach. The pioneering FlowNet of Dosovitskiy et al. [8] showed that motion estimation can be learned end-to-end with a CNN. Later works [18,11] elaborated on this concept and provided multiscale and multistep approaches. The FlowNet2 by Ilg et al. [11] yields state-of-the-art accuracy but is orders of magnitudes faster than traditional methods. We use this network as a building block for end-to-end training of a video super-resolution network.

3 Video super-resolution with patch-based training

In this section we revisit the work from Kappeler et al. [12], which applies network-external motion compensation and then extends the single-image SRCNN [7] to operate on multiple frames. This approach is shown in Figure 1(a).



(a) Architecture as proposed by Kappeler et al. [12]



(b) Architecture with integrated FlowNet2-SD from [11]

Fig. 1: Video super-resolution architectures used by the basic models tested in this paper. Optical flow is estimated from the center to the outer frames using either an external method or a CNN. The flow is used to warp all frames to the center frame. The frames are then input into the VSR network [12]. The complete network in (b) can be trained end-to-end including the motion estimation.

Kappeler et al. [12] compare different numbers of input frames and investigate early and late fusion by performing the concatenation of features from the different frames after different layers. They conclude that fusion after the first convolution works best. Here, we use this version and furthermore stick to three input frames and an upsampling factor of four throughout the whole paper.

We performed an analysis of their code and model. The results are given in the first row of Table 1. Using their original code, we conducted an experiment, where we replaced the three frames from the image sequence by three times the same center frame (column 4 of Table 1), which corresponds to the information

only from single-image super-resolution. We find that on the Myanmar validation set the result is still much better than SRCNN [7] but only marginally worse than VSR [12] on real video information. Since except for a concatenation there is no difference between the VSR [12] and SRCNN [7] architectures, this shows that surprisingly the improvement is mainly due to training settings of VSR [12] rather than the usage of multiple frames.

For training and evaluation, Kappeler et al. [12] used the publicly available Myanmar video [1]. We used the same training/validation split into 53 and 6 scenes and followed the patch sampling from [12]. However, the publicly available data has changed by that the overlaid logo at the bottom right corner from the producing company is now bigger than before. Evaluating on the data with the different logo gives much worse results (row 2 of Table 1), while when the logo is cropped off (column 3 of Table 1), results are comparable. The remaining difference stems from a different implementation of the warping operation¹. However, when we retrained the approach with our implementation and training data (row 3 of Table 1), we achieved results very close to Kappler et al. [12].

To further investigate the effects of motion compensation, we retrained the approach using only the center frame, the original frames, and frames motion compensated using FlowNet2 [11] and FlowNet2-SD [11] in addition to the method from Drulea [9]. For details we refer to the supplemental material. Again we observed that including or excluding motion compensation with different optical flow methods has no effect on the Myanmar validation set. We additionally evaluated on the commonly used Videoset4 dataset [16,12]. In this case we do see a PSNR increment of 0.1 with Drulea [9] and higher increment of 0.18 with FlowNet2 [11] when using motion compensation. The Videoset4 dataset includes larger motion and it seems that there is some small improvement when larger motion is involved. However, the effect of motion compensation is still very small when compared to the effect of changing other training settings.

4 Video super-resolution with image-based training

In contrast to Kappeler et al., we combine motion compensation and super-resolution in one network. For motion estimation, we used the FlowNet2-SD variant from [11]. We chose this network, because FlowNet2 itself is too large to fit into GPU memory besides the super-resolution network and FlowNet2-SD yields smooth flow predictions and accurate performance for small displacements. Figure 1(b) shows the integrated network. For the warping operation, we use the implementation from [11], which also allows a backward pass while training. The combined network is trained on complete images instead of patches. Thus, we repeated our experiments from the previous section for the case of image-based training. The results are given in Table 2. In general, we find that image-based processing yields much higher PSNRs than patch-based processing. Detailed

¹ We use the implementation from [11]; it differs from [12] in that it performs bilinear interpolation instead of bicubic.

comparison of the network and training settings for both variants can be found in the supplemental material.

Dataset/Model	SRCNN [7]	VSR [12] (cropped)	VSR [12] (only center)	VSR [12] (no warp.)
Myanmar validation from [12]	31.26	31.81	32.95	31.71
Myanmar validation (ours)	31.30	31.30	32.88	31.23
Myanmar validation (ours), retrained		31.81	32.76	31.74
				31.77

Table 1: Analysis of Kappeler et al. [12] on the different versions of the Myanmar dataset. Numbers show the PSNR in dB. The first row is with the original code and test data from [12], while the second and third row are with our reimplementation and the new test data that was recently downloaded. The third column shows results when the logo area is cropped off. Fourth and fifth columns show the PSNR when motion compensation is disabled during testing, by using only the center frame or the original frames without warping. There is no significant improvement by neither the use of multiple frames nor by the use of optical flow.

Network	SRCNN [7]	VSR [12] only center	VSR [12] no warp.	VSR [12] Drulea [9]	VSR [12] FN2-SD [11]	VSR [12] FN2-SD [11] joint
Motion Compensation	-					
Myanmar validation (ours)	32.42	32.41	32.55	32.60	32.62	32.63
Videoset4	24.63	24.66	24.79	24.91	25.12	25.21

Table 2: PSNR scores from Myanmar validation (ours) and Videoset4 for image-based training. For each column of the table we trained the architecture of [7] and [12] by applying convolutions over the complete images. We used different types of motion compensation for training and testing (FN2-SD denotes FlowNet2-SD). For Myanmar, motion compensation still has no significant effect. However, on Videoset4 an effect for motion compensation using Drulea’s method [9] is noticeable and is even stronger for FlowNet2-SD[11].

Table 2 shows that motion compensation has no effect on the Myanmar validation set. For Videoset4 there is an increase of 0.12 with motion compensation using Drulea’s method [9]. For FlowNet2 the increase of 0.42 is even bigger. Since FlowNet2-SD is completely trainable, it is also possible to refine the optical flow for the task of video super-resolution by training the whole network end-to-end with the super-resolution loss. We do so by using a resolution of 256×256 to enable a batch size of 8 and train for 100k more iterations. The results from Table 2 again show that for Myanmar there is no significant change. However, for Videoset4 the joint training further improves the result by 0.1 leading to a total PSNR increase of 0.52.

We show a qualitative evaluation in Figure 2. On the enlarged building, one can see that bicubic upsampling introduces some smearing across the windows. This effect is also present in the methods without motion compensation and in the original VSR [12] with motion compensation. When using image-based trained models, the effect is successfully removed. Motion compensation with FlowNet2 [11] seems to be marginally sharper than motion compensation with Drulea [9]. We find that the joint training reduces ringing artifacts; an example is given in the supplemental material.

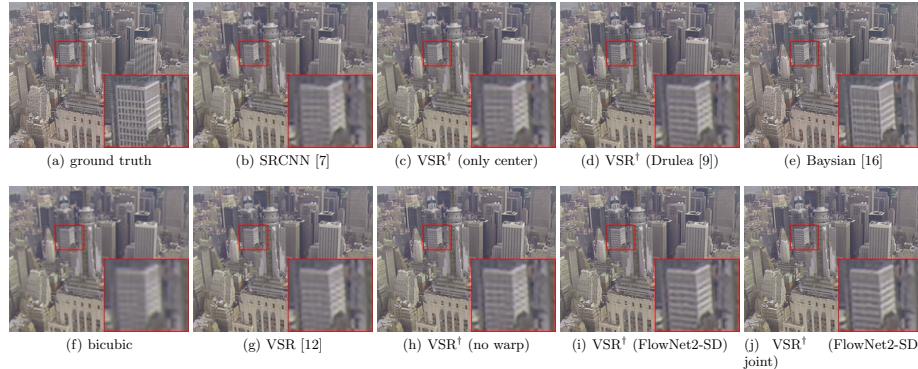


Fig. 2: Comparison of existing super-resolution methods to our trained models. \dagger indicates models retrained by us using image-based training. Note that b) and g) are patch-based, while c), d), e), h), i) and j) are image-based.

5 Combined Warping and Upsampling Operation

The approach of Kappeler et al. [12] and the VSR architecture discussed so far follow the common practice of first upsampling and then warping the images. Both operations involve an interpolation during which image information is lost. Therefore, we propose a joint operation that performs upsampling and backward warping in a single step, which we name Joint Upsampling and Backward Warping (*JUBW*). This operation does not perform any interpolation at all, but additionally outputs sub-pixel distances and leaves finding a meaningful interpolation to the network itself. Let us consider a pixel p and let x_p and y_p denote the coordinates in high resolution space, while x_p^s and y_p^s denote the source coordinates in low resolution space using high resolution flow estimations (u_p, v_p) is computed according to the following equation:

$$\begin{pmatrix} x_p^s \\ y_p^s \end{pmatrix} = \frac{1}{\alpha} \begin{pmatrix} x_p + u_p + 0.5 \\ y_p + v_p + 0.5 \end{pmatrix} - \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}, \quad (1)$$

where $\alpha = 4$ denotes the scaling factor and subtraction/addition of 0.5 places the origin at the top left corner of the first pixel. Then the warped image is computed as:

$$I_w(p) = \begin{cases} I(\lfloor x_p^s \rfloor, \lfloor y_p^s \rfloor) & \text{if } \lfloor x_p^s \rfloor, \lfloor y_p^s \rfloor \text{ is inside } I, \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where $\lfloor \cdot \rfloor$ denotes the round to nearest operation. Note, that no interpolation between pixels is performed. The operation then additionally outputs the following distances per pixel (see Figure 3 for illustration):

$$\begin{pmatrix} d_p^x \\ d_p^y \end{pmatrix} = \begin{pmatrix} \lfloor x_p^s \rfloor - x_p^s \\ \lfloor y_p^s \rfloor - y_p^s \end{pmatrix} \text{ if } \lfloor x_p^s \rfloor, \lfloor y_p^s \rfloor \text{ is inside } I \text{ and } \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ otherwise.} \quad (3)$$

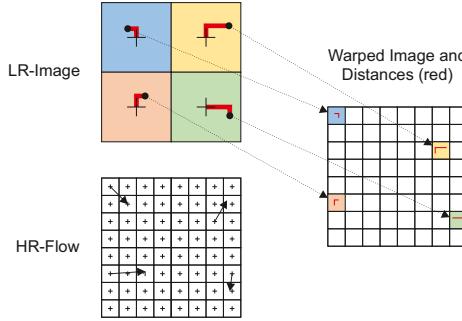


Fig. 3: Illustration of the Joint Upsampling and Backward Warping operation (JUBW). The output is a dense image (left sparse here for illustration purposes) and includes x/y distances of the source locations to the source pixel centers.

We also implemented the joint upsampling and forward warping operation from Tao et al. [23] for comparison and denote it as SPMC-FW. Contrary to our operation, SPMC-FW still involves two types of interpolation: 1.) subpixel-interpolation for the target position in the high resolution grid and 2.) interpolation between values if multiple flow vectors point to the same target location. For comparison, we replaced the architecture from the previous section by the encoder-/decoder part from Tao et al. [23] (which we denote here as SPMC-ED). We also find that this architecture itself performs better than SRCNN [7]/VSR [12] on the super-resolution only task (see supplementary material for details). The resulting configuration is shown in Figure 4. Furthermore, we also extended the training set by downloading Youtube videos and downampling them to create additional training data. The larger dataset comprises 162k images and we call it MYT.

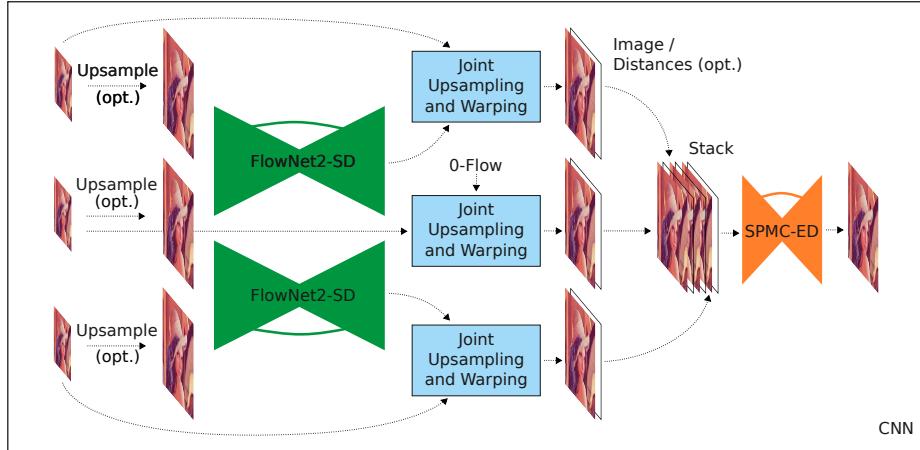


Fig. 4: Network setup with FlowNet2-SD and joint upsampling and warping operation (JUBW or SPMC-FW). Upsampling before feeding into FlowNet2-SD happens only for JUBW. The output of the upsampling and warping operation is stacked and then fed into the SPMC-ED network.

	SPMC original [23]	SPMC-FW			JUBW			
		ours	only center	joint	ours	no dist.	only center	joint
Myanmar (ours)	-	32.90	32.45	33.05	33.13	33.02	32.55	32.69
Videoset4	25.52	25.68	24.94	25.62	25.85	25.74	24.96	25.09

Table 3: PSNR values for different joint upsampling and warping approaches. The first column shows the original results from Tao et al. [23] using the SPMC upsampling, forward warping, and the SPMC-ED architecture with an LSTM unit. Columns two to four show our reimplementations of the SPMC-FW operation [23] without an LSTM unit. Columns five to eight show our joint upsampling and backward warping operation with the same encoder-decoder network on top. With *ours* we denote our implementation according to Figure 4. In *only center* we input zero-flows and the duplicated center image three times (no temporal information). The entry *joint* includes joint training of FlowNet2-SD and the super-resolution network. For columns two to eight, the networks are retrained on MYT and tested for each setting respectively.

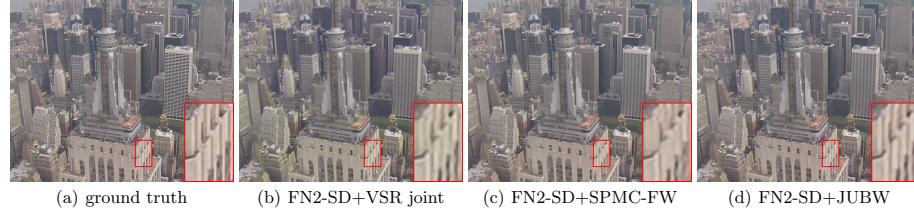


Fig. 5: Examples of a reconstructed image from Videoset4 using different warping methods. FN2-SD stands for FlowNet2-SD. Clearly using JUBW yields sharper and more accurate reconstruction of the estimated frames compared to SPMC-FW [23] and the best VSR [12] result.

Results are given in Table 3. First, we note that our feed-forward implementation of FlowNet2-SD with SPMC-ED, which simply stacks frames and does not include an LSTM unit, outperforms the original recurrent implementation from Tao et al. [23]. Second, we see that our proposed JUBW operation generally outperforms SPMC-FW. We again performed experiments where we excluded temporal information, by inputting zero flows and duplicates of the center image. We now observe that including temporal information yields large improvements and increases the PSNR by 0.5 to 0.9. In contrast to the previous sections, we see such increase also for the Myanmar dataset. This shows that the proposed motion compensation can also exploit small motion vectors. The qualitative results in Fig. 5 confirm these findings.

Including the sub-pixel distance outputs from JUBW layer to enable better interpolation to the network leads to a smaller improvement than expected. Notably, without these distances the JUBW operation degrades to a simple nearest neighbor upsampling and nearest neighbor warping, but it still outperforms SPMC-FW. We conclude from this that one should generally avoid any kind of interpolation and leave it to the network. Finally, fine-tuning FlowNet2 on the video super-resolution task decreases the PSNR in some cases and does not provide the best results. We conjecture that this is due to the nature of optimization of the gradient through the warping operation, which is based on the reconstruction error and is prone to local minima.

6 Conclusions

In this paper, we performed an evaluation of different video super-resolution approaches using CNNs including motion compensation. We found that the common practice of patch-based training and upsampling and warping separately yields almost no improvement when comparing the video super-resolution setting against the single-image setting. We obtained a significant improvement over prior work by replacing the patch-based approach by a network that analyzes the whole image. As a remedy for the lacking standard motion compensation, we proposed a joint upsampling and backward warping operation and combined it with FlowNet2-SD [11] and the SPMC-ED [23] architecture. This combination out-

performs all previous work on video super-resolution. In conclusion, our results show that: 1.) we can achieve the same or better performance with a formulation as a feed-forward instead of a recurrent network; 2.) performing joint upsampling and backward warping with no interpolation outperforms joint upsampling and forward warping and the common backward warping with interpolation; 3.) including sub-pixel distances yields a small additional improvement; and 4.) joint training with FlowNet2-SD so far does not lead to consistent improvements and we leave a more detailed analysis to future work.

Acknowledgements

We acknowledge the DFG Grant BR-3815/7-1.

References

1. Myanmar 60p, Harmonic Inc. (2014), [online]. Available: <https://www.harmonicinc.com/resources/videos/4k-video-clip-center>
2. Babacan, S.D., Molina, R., Katsaggelos, A.K.: Variational bayesian super resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 20(4), 984–999 (Apr 2011)
3. Caballero, J., Ledig, C., Aitken, A.P., Acosta, A., Totz, J., Wang, Z., Shi, W.: Real-time video super-resolution with spatio-temporal networks and motion compensation. *CoRR* abs/1611.05250 (2016), <http://arxiv.org/abs/1611.05250>
4. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2004)
5. Chao Dong, Chen Change Loy, X.T.: Accelerating the super-resolution convolutional neural network. In: *European Conference on Computer Vision (ECCV)* (2016)
6. Cheng, M.H., Lin, N.W., Hwang, K.S., Jeng, J.H.: Fast video super-resolution using artificial neural networks. In: *International Symposium on Communication Systems, Networks Digital Signal Processing (CSNDSP)*. pp. 1–4 (July 2012)
7. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 38(2), 295–307 (Feb 2016)
8. Dosovitskiy, A., Fischer, P., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., Smagt, P., Cremers, D., Brox, T.: FlowNet: Learning optical flow with convolutional networks. In: *IEEE International Conference on Computer Vision (ICCV)* (2015)
9. Drudea, M., Nedevschi, S.: Total variation regularization of local-global optical flow. In: *IEEE Conference on Intelligent Transportation Systems (ITSC)*. pp. 318–323 (Oct 2011)
10. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. *IEEE Computer Graphics and Applications (ICGA)* 22(2), 56–65 (Mar 2002)
11. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (July 2017)

12. Kappeler, A., Yoo, S., Dai, Q., Katsaggelos, A.K.: Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging (TCI)* 2(2), 109–122 (June 2016)
13. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 1646–1654 (June 2016)
14. Kim, J., Lee, J.K., Lee, K.M.: Deeply-recursive convolutional network for image super-resolution. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* abs/1511.04491 (2016)
15. Ledig, C., Theis, L., Huszar, F., Caballero, J., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
16. Liu, C., Sun, D.: On bayesian adaptive video super resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 36(2), 346–360 (Feb 2014)
17. Ma, Z., Liao, R., Tao, X., Xu, L., Jia, J., Wu, E.: Handling motion blur in multi-frame super-resolution. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 5224–5232 (June 2015)
18. Ranjan, A., Black, M.J.: Optical flow estimation using a spatial pyramid network. CoRR abs/1611.00850 (2016), <http://arxiv.org/abs/1611.00850>
19. Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016)
20. Song, B.C., Jeong, S.C., Choi, Y.: Video super-resolution algorithm using bi-directional overlapped block motion compensation and on-the-fly dictionary training. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)* 21(3), 274–285 (Mar 2011)
21. Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 2432–2439 (June 2010)
22. Takeda, H., Milanfar, P., Protter, M., Elad, M.: Super-resolution without explicit subpixel motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 18(9), 1958–1975 (Sep 2009)
23. Tao, X., Gao, H., Liao, R., Wang, J., Jia, J.: Detail-revealing deep video super-resolution. CoRR abs/1704.02738 (2017), <http://arxiv.org/abs/1704.02738>
24. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2008)
25. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 19(11), 2861–2873 (Nov 2010)

Supplemental Material for End-to-End Learning of Video Super-Resolution with Motion Compensation

Osama Makansi, Eddy Ilg, and Thomas Brox

Department of Computer Science, University of Freiburg

1 Computation of PSNR values

For all our evaluation results, the reported PSNR values are computed using only the Y channel of the estimated YCbCr image. In case of RGB images, we first convert to YCbCr color space and then compute on the Y channel. For all experiments using the SRCNN [1] or VSR [5] architecture, we follow [5] and for technical reasons crop away 12 pixels of the boundary from the estimated high-resolution images before computing PSNR values.

2 Displacement magnitudes

We have noted that improvements using motion compensation are generally smaller on Myanmar than on Videoset4. In Table 1, we compute the average motion magnitudes of the datasets and note that the displacements are also generally smaller in the Myanmar validation set.

Dataset	Avg. Mag.
Myanmar training	1.50px
Myanmar validation	0.43px
Videoset4	1.29px

Table 1: Average motion magnitudes computed using FlowNet2 [4]. The numbers show that the Myanmar validation set has the smallest displacements.

3 Video super-resolution with patch-based training

Using patch-based training, we retrain and evaluate SRCNN [1] and VSR [5] using different kind of motion compensations. However, the resulting PSNR scores in Table 2 are all similar and we conclude that motion compensation on Myanmar has no effect. We also evaluate on Videoset4 (Table 3) and there see a small increment of 0.18 for FlowNet2 [4] and FlowNet2-SD [4].

Arch.	Trained on \ Tested on					
		only center	no warp	Drulea [2]	FlowNet2-SD [4]	FlowNet2 [4]
SRCNN	only center	31.62	-	-	-	-
VSR	only center	31.76	-	-	-	-
	no warp	31.80	31.83	-	-	-
	Drulea [2]	31.77	31.74	31.81	-	-
	FlowNet2-SD [4]	31.75	31.75	31.79	31.77	-
	FlowNet2 [4]	31.76	31.76	31.80	31.78	31.79

Table 2: PSNR scores for patch-based video superresolution on the Myanmar validation set. We retrained the architecture of [5] using only the center frames (replicated three times), original images, and motion compensated frames. One can observe that all scores are nearly the same and motion compensation on the Myanmar validation set has no effect over providing original images or even only the center image.

Motion compensation during training and testing	Videoset4 PSNR
only center	24.60
no warp	24.59
Drulea [2]	24.69
FlowNet2-SD [4]	24.77
FlowNet2 [4]	24.77

Table 3: Evaluation of the different retrained VSR models from Table 2 on Videoset4. Motion compensation shows a small performance improvement.

Setting	Patch-based	Image-based
Learning rate	$1e - 05$	$1e - 05$
Learning rate policy	fixed	multistep [†]
Momentum	0.9	0.9
Weight decay	0.0005	0.0004
Batch size	240	2
Input resolution	36×36	960×540
Image pixels in batch	311k	1M
Training iterations	200k	300k
Training time	7 hours	32 hours

Table 4: Different settings of patch- and image-based training. Settings are very similar, except that the number of pixels and training time in image-based training are larger. Note that the number of pixels is also further boosted much more by sliding the convolutions from the VSR architecture over the entire images with a stride of one. [†]multiplied by 0.5 every 50k iterations.

4 Video super-resolution with image-based training

We perform the same set of experiments as in the last section for image-based training. Comparing Table 2 to Table 5, we find that PSNRs are generally around 1 point higher. We also provide all the training settings in Table 4. Image-based training in general processes more training data and sees a lot of similar data during training by sliding the convolutions over an entire image. Motion compensation on Myanmar (Table 5) still seems to have little effect, while motion compensation on Videoset4 does show better PSNR values (Table 6).

Arch.	Tested on		only center	no warp	Drulea [2]	FlowNet2-SD [4]	FlowNet2 [4]
	Trained on						
VSR	only center		32.41	-	-	-	-
	no warp		32.38	32.55	-	-	-
	Drulea [2]		32.37	32.26	32.60	-	-
	FlowNet2-SD [4]		32.35	32.37	32.58	32.62	-
	FlowNet2 [4]		32.37	32.36	32.61	32.61	32.63

Table 5: PSNR scores from Myanmar validation (ours). We now train the architecture of [5] by applying it as a convolution over the complete images. We again evaluate using only the center frame, original images and differently motion compensated frames. One can observe that scores are significantly better compared to the patch-based training, but motion compensation on the Myanmar validation set still has negligible effect compared to training on original frames.

Training input	PSNR
only center	24.66
no warp	24.79
Drulea [2]	24.91
FlowNet2-SD [4]	25.12
FlowNet2 [4]	25.13

Table 6: Evaluation of the different image-based models on Videoset4. Motion compensation in this case also shows a performance improvement.

5 Joint Training

Since the FlowNet2-SD [4] is completely trainable, we can refine the optical flow for the task of video super-resolution by training the whole network end-to-end with the super-resolution loss. This potentially allows the optical flow estimation to focus on aspects that are most relevant for the super-resolution task. As an initialization we took the VSR network trained on FlowNet2-SD [4] from the last section and used the same settings from Table 4, but now cropped the images to a resolution of 256×256 to enable a batch size of 8. We then trained for 100k more iterations. The result is given in Table 7 and Figures 1(b) and 1(e). We cannot see the flow itself improve, but we see a small improvement in the PSNR value on Videoset4 and from the images one can observe that the ringing artifacts disappear.

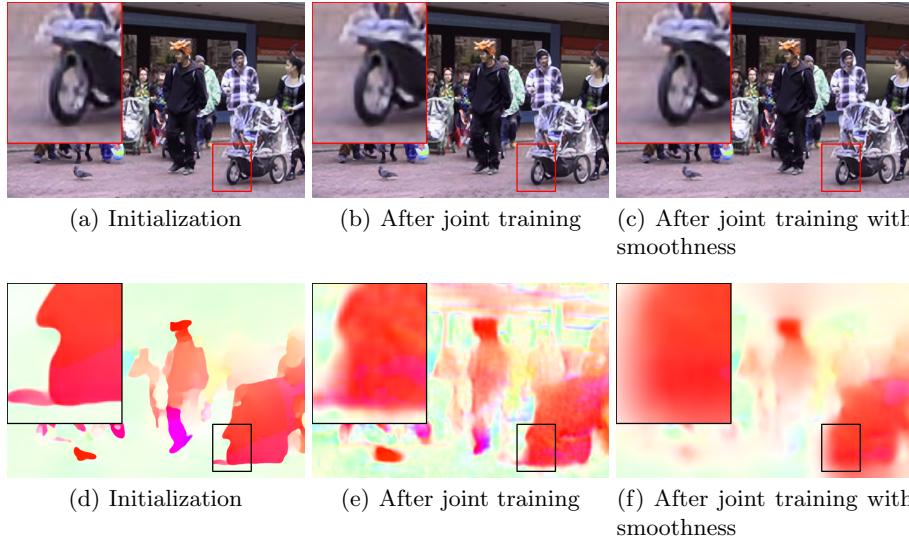


Fig. 1: Example super-resolved image after training FlowNet2-SD [4] with VSR (a and d) jointly (b and e) and including a smoothness constraint (c and f).

In Figure 1(e), one can observe that many image details become flow artifacts. This is due to nature of the gradient through the warping operation; it corrects the flow vector to the best directly neighboring pixel, which is in most cases a local minimum. Following [3], we add a regularization loss that penalizes deviations from smoothness in the optical flow field, weighted with an image edge-aware term:

$$\mathcal{L}_R = \sum_{i,j} \left(e^{-\|\partial_x I_{i,j}\|} (|\partial_x u| + |\partial_x v|) + e^{-\|\partial_y I_{i,j}\|} (|\partial_y u| + |\partial_y v|) \right) \quad (1)$$

Test set	After initialization	After joint training	After joint training with smoothness
Myanmar validation	32.62	32.63	32.61
VideoSet4	25.12	25.21	25.19

Table 7: Evaluation of refining FlowNet2-SD [4] on the super-resolution task.

where I is the first image. i, j is a pixel location and u, v are the x, y components of the flow vector. The results of training with this additional smoothness term are given in Table 7 and Figures 1(c) and 1(f). The flow shows less artifacts than Figure 1(e), but compared to Figure 1(b) some very slight ringing artifacts still remain.

6 Evaluating Architectures and Datasets

In first part of the paper, the architecture from Dong et al. [1] adapted to video super-resolution by Kappeler et al. [5] and the Myanmar training dataset were used. Here, we investigate the effect of architectures and training datasets. We extended the Myanmar training set by more high-resolution videos that we downloaded from Youtube. The resulting dataset has 162k frames of resolution 960×540 and we named it MYT. We evaluate and compare the SRCNN [1], the FlowNet2-SD [4] (here used for super-resolution, not flow) and the encoder-/decoder part of the architecture from Tao et al. [6] (SPMC-ED) for single image super-resolution on the old and new datasets. The results are given in Table 8.

	SRCNN [1] trained on Myanmar training (ours)	SRCNN [1] trained on MYT	FlowNet2-SD [4] trained on MYT	SPMC-ED [6] trained on MYT
Myanmar validation (ours)	32.42	31.98	31.47	32.63
Videoset4	24.63	24.70	24.93	25.07
Number of parameters	57K	57K	14M	491K

Table 8: PSNR values for different architectures and training datasets tested for single-image super-resolution.

One can observe that SRCNN [1] tends to overfit on the Myanmar dataset. The much deeper FlowNet2-SD [4] architecture performs worse on Myanmar, but can generalize better to Videoset4. The size of SPMC-ED [6] is between the former two and we observe that it performs best on Myanmar and also for generalization to Videoset4. It clearly gives better results than SRCNN [1] and for this reason we also use it for the final network in the paper.

References

1. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 38(2), 295–307 (Feb 2016)
2. Drulea, M., Nedevschi, S.: Total variation regularization of local-global optical flow. In: *IEEE Conference on Intelligent Transportation Systems (ITSC)*. pp. 318–323 (Oct 2011)
3. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. *CoRR* abs/1609.03677 (2016), <http://arxiv.org/abs/1609.03677>
4. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (July 2017)
5. Kappeler, A., Yoo, S., Dai, Q., Katsaggelos, A.K.: Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging (TCI)* 2(2), 109–122 (June 2016)
6. Tao, X., Gao, H., Liao, R., Wang, J., Jia, J.: Detail-revealing deep video super-resolution. *CoRR* abs/1704.02738 (2017), <http://arxiv.org/abs/1704.02738>