Stereo Super-resolution via a Deep Convolutional Network

Junxuan Li[1] Shaodi You[1,2] Antonio Robles-Kelly[1,2]
[1] College of Eng. and Comp. Sci., The Australian National University, Canberra ACT 0200, Australia
[2] DATA61 - CSIRO, Tower A, 7 London Circuit, Canberra ACT 2601, Australia

***Abstract*—In this paper, we present a method for stereo super-resolution which employs a deep network. The network is trained using the residual image so as to obtain a high resolution image from two, low resolution views. Our network is comprised by two deep sub-nets which share, at their output, a single convolutional layer. This last layer in the network delivers an estimate of the residual image which is then used, in combination with the left input frame of the stereo pair, to compute the super-resolved image at output. Each of these sub-networks is comprised by ten weight layers and, hence, allows our network to combine structural information in the image across image regions efficiently. Moreover, by learning the residual image, the network copes better with vanishing gradients and its devoid of gradient clipping operations. We illustrate the utility of our network for image-pair super-resolution and compare our network to its non-gradient trained analogue and alternatives elsewhere in the literature.**

***Index Terms*—stereo super-resolution, convolutional neural network, residual training**

I. Introduction

Image super-resolution is a classical problem which has found application in areas such as video processing [1], light field imaging [2] and image reconstruction [3].

Given its importance, super-resolution has attracted ample attention in the image processing and computer vision community. Super-resolution approaches use a wide range of techniques to recover a high-resolution image from low-resolution imagery. Early approaches to super-resolution are often based upon the rationale that higher-resolution images have a frequency domain representation whose higher-order components are greater than their lower-resolution analogues. Thus, methods such as that in [4] exploit the shift and aliasing properties of the Fourier transform to recover a super-resolved image. Kim *et al.* [5] extended the method in [4] to settings where noise and spatial blurring are present in the input image. In a related development, in [6], super-resolution in the frequency domain is effected using Tikohonov regularisation. In [7], the motion and the higher-resolution image are estimated simultaneously using the EM algorithm.

Other methods, however, adopt an interpolation approach to the problem, whereby the lower resolution input image is related to the higher-resolved one by a sparse linear system. These methods profit from the fact that a number of statistical techniques can be naturally adapted to solve the problem in hand. These include maximum likelihood estimation [8] and wavelets [9]. These methods are somewhat related to the projection onto convex sets (POCS) approach [10]. This is a set-based image restoration method where the convex sets are used to constrain the super-resolution process.

The methods above are also related to example-based approaches, where super-resolution is effected by *aggregating* multiple frames with complementary spatial information. Baker and Kanade [11] formulate the problem in a regularisation setting where the examples are constructed using a pyramid approach. Protter *et al.* [12] used block matching to estimate a motion model and use exemplars to recover super-resolved videos. Yang *et al.* [13] used sparse coding to perform super-resolution by learning a dictionary that can then be used to produce the output image, by linearly combining learned exemplars.

Moreover, the idea of super-solution "by example" can be viewed as hinging on the idea of learning functions so as to map a lower-resolution image to a higher-resolved one using exemplar pairs. This is right at the centre of the philosophy driving deep convolutional networks, where the net is often considered to learn a non-linear mapping between the input and the output. In fact, Dong *et al.* present in [14] a deep convolutional network for single-image super-resolution which is equivalent to the sparse coding approach in [13], [15]. In a similar development, Kim *et al.* [16] present a deep convolutional network inspired by VGG-net [17]. The network in [16] is comprised by 20 layers so as to exploit the image context across image regions. In [18], a multi-frame deep network for video super-resolution is presented. The network employs motion compensated frames as input and single-image pre-training.

Here, we present a deep network for stereo super-resolution which takes two low-resolution, paraxially shifted frames and delivers, at output, a super-resolved image. The network is somewhat reminiscent to those above, but there are two notable differences. Firstly, as shown in Figure 1, we use two networks in tandem, one for each of the input stereo frames, and then combine them at the last layer. This contrasts with other networks in the literature where the low resolution frames are concatenated or aggregated at input. Secondly, instead of the typical loss function used in deep nets, we employ a residual learning scheme [19]. This residual scheme is not only known to deal with the vanishing gradients well but has also been suggested it improves convergence.

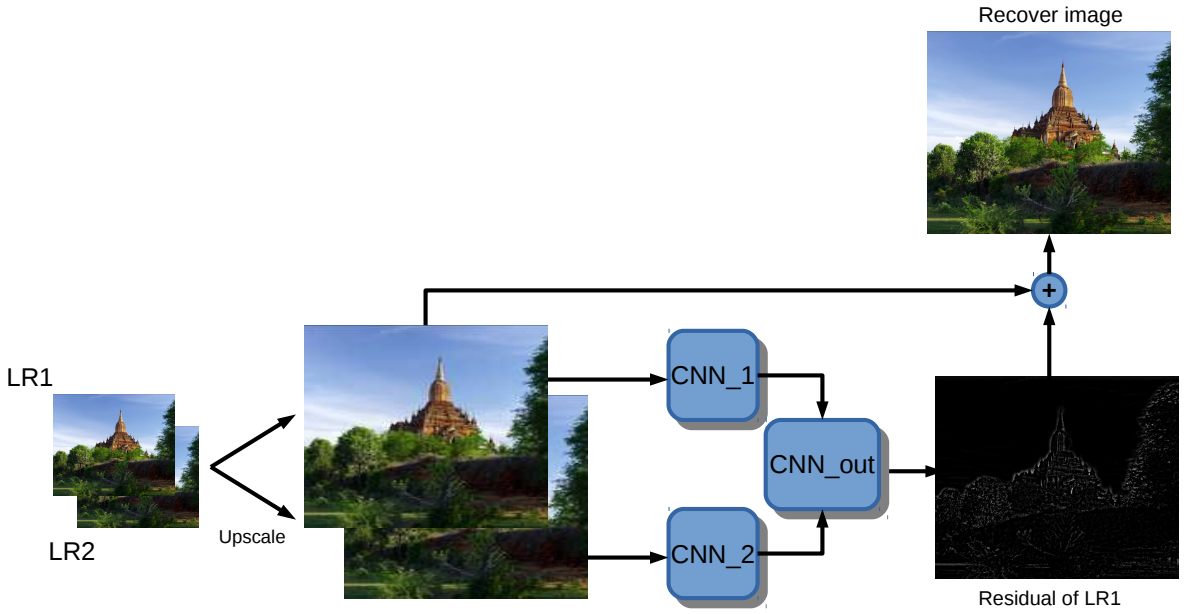

Fig. 1. Simple diagram showing the structure of our network. At input, the low resolution image pair is upscaled and used as input to the two sub-nets (one for each view). The output of these sub-networks is then concatenated to be used as the input to another network which, in turn, combines these to obtain the residual image. The residual image is then added to the up-sampled left input frame to obtain the super-resolved image.

II. Deep Network for Stereo Super-resolution

A. Architecture

As mentioned above, and shown in Figure 1, we have used two sub-networks for each of the stereo pairs and then a single output convolutional layer which delivers the image residual. This residual is then used, in conjunction with the left input frame, to obtain the super-resolved image. This can be appreciated in more detail in Figure 2, where the two input low resolution images, denoted LR1 and LR2 are then resized to their target output size. These two re-sized images are then fed into each of the two sub-networks.

Each of these two sub-networks are 10 layers deep. Each layer is comprised by a convolution operation with 32 filters of size 3×3 followed by batch normalization and a linear rectifier unit (ReLU). In our network we have not included gradient clipping. Note that these sub-networks are somewhat reminiscent to that in [16]. Indeed, nonetheless the filters are 3×3, the layer can still exploit the image context across image regions which are much larger than the filters themselves. In this manner, the network can employ contextual information to obtain a super-resolved image.

The two sub-networks then deliver an output of size $W \times H \times 32$, where W and H are the width and height of the upscaled images. These two outputs are then concatenated to obtain a $W \times H \times 64$ tensor which is then used as input to the last convolutional layer of our network. This layer employs a single 5×5 filter to obtain the image residual. This layer still employs batch normalisation but, unlike the other layers in the network, lacks a rectification unit.

B. Residual Learning

As mentioned earlier, here we use a residual learning approach to train our network. This concept was introduced in [19] as a powerful tool for dealing with the vanishing gradients problem [20]. It was later applied to single image super-resolution in [16]. In [16], the authors also note that the application of the residual appears to have a positive impact in the training of the network, which, as they report, enhances the convergence rate of the learning process.

Our target residual image R, is defined using the difference between the low resolution upsampled image $\hat{I}$ and the high resolution frame I from the training set, *i.e.* $R = I - \hat{I}$. The residual is used to compute an $L2$ loss function of the form

$$L = \frac{1}{2} \mid R(u) - \hat{R}(u) \mid_2^2 \tag{1}$$

where, as usual, $\mid \cdot \mid_2^2$ is the squared $L2$ norm and $R(u)$ and $\hat{R}(u)$ are the target residual and that delivered by our network for the pixel u in the imagery.

In this way, the value of the pixel u for the high resolution image I^* can be computed in a straightforward manner using the expression

$$I^*(u) = \hat{I}(u) + \hat{R}(u) \tag{2}$$

III. Experiments

In this section, we present a qualitative and quantitative evaluation of our method and compare against alternatives elsewhere in the literature. The section is organised as follows. We commence by introducing the datasets we have used for training and testing. Later on in the section we elaborate upon the implementation of our method. We conclude the section by

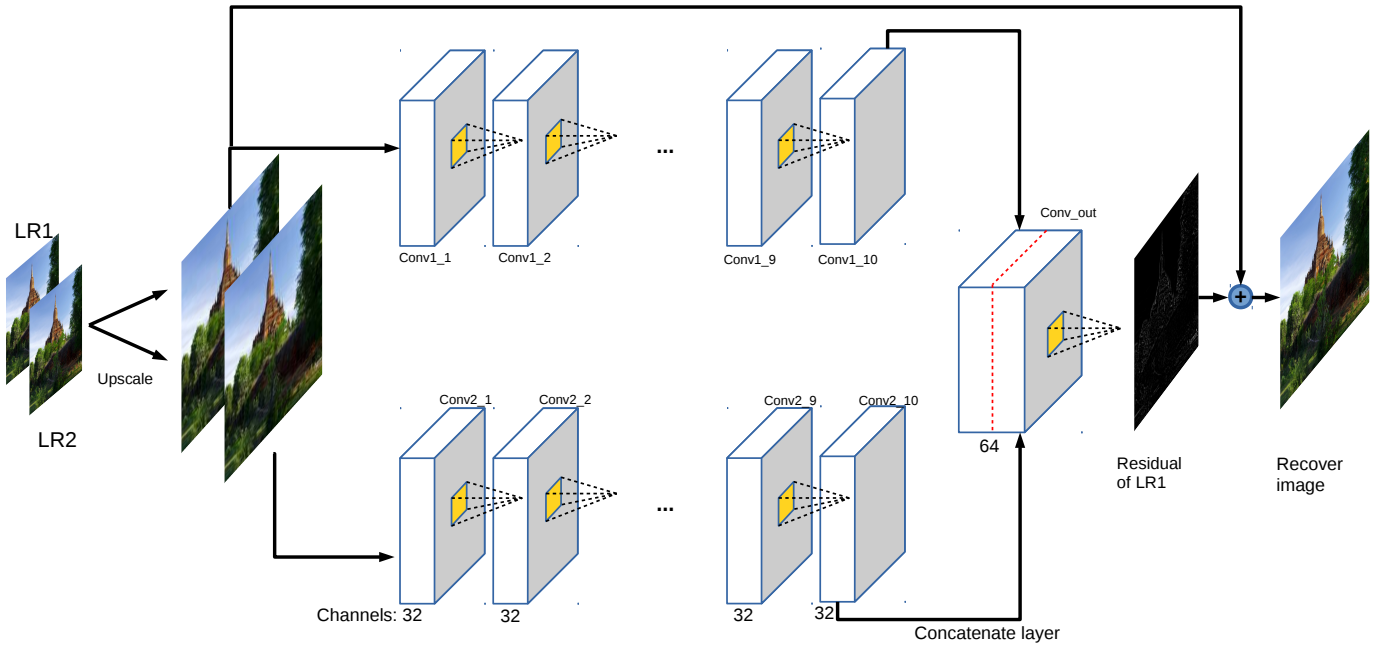

Fig. 2. Detailed diagram of our network architecture. Each of the blocks labelled Conv1$_i$, where i denotes the index of the network layer consist of a convolution, batch normalization and ReLU. Each of the weight layers for the two sub-networks are comprised by 32 filters of size 3×3. The last layer, denoted Conv_out is composed of a single 5×5 filter.

presenting the results yielded by our network and comparing these against alternative approaches.

A. Datasets

1) Training: For purposes of training, we have used the Harmonic Myanmar 60p footage. The dataset is publicly available [1]. This contrasts with other methods elsewhere in the literature. For instance, the authors in [14] employ a large set of images from ImageNet while the method in [16] uses 382 images and data augmentation through rotation or flip to obtain the training set.

The choice of our training set is mainly motivated by the size of our network. Note that our network, with its ten layers per stereo image and the final common output layer has a large number of parameters to train. Thus, we have chosen to employ a video dataset where stereo pairs are comprised of consecutive frames. Further, the dataset was also used for training and testing the VSRNET [18]. The main difference to note, however, is that VSRNET [18] is a video, *i.e.* multi-frame, network rather than a stereo vision one and, therefore, typically employs 5 frames to recover the super-resolved image.

Here, we have used 30 scenes comprising 15000 frames from the training video and taken 27 scenes for training and the remaining ones for validation. Note that each frame is 4K resolution, *i.e.* 3840×2160 px. This also contrasts with other methods elsewhere where the common practice is to use training sets with typical resolutions of 720×1024px. For example, VideoSet4 [22] employs imagery of 720×576 px whereas VideoSet14 [23] uses images with resolution of 800×800 px.

The above details regarding resolution are important since they motivate down scaling the 4K video used for training by a factor of 4 to 960×540 px. Following [14], we convert the video frames to the YCbCr colorspace [24] and employ the luminance, *i.e.* the Y channel, for training. Moreover, we note that, in the video frames used for training, there are large image regions where the structure conveyed by the imagery is very scarce and, hence, they're contribution to the residual is negligible. This is straightforward to note in sky or flat surfaces where the low and high resolution imagery are expected to have minimal variations with respect to one another. Therefore, we have cropped the images into non-overlapping small patches of size 96×54 and chosen those with clear structural artifacts, such as edges and corners. This yields a set of 330000 96×54 px patches for training.

2) Testing: Here, we have followed [22] and used the four test scenes, *i.e.* City, Walk, Calendar and Foliage, employed by VideoSet4 for purposes of performance evaluation and testing. Each of these videos have a resolution of 720×576 px.

At input, each pair of input video frames are converted to the YCbCr colorspace [24]. Once the luminance is in hand, we use our trained network for super-resolving the Y channel. The luminance is then used in combination with the chrominance, *i.e.* the CbCr channels, to obtain the final trichromatic image in the sRGB space [24]. In all our tests, we have referenced our

[1]The dataset can be downloaded at https://www.harmonicinc.com/free-4k-demo-footage/

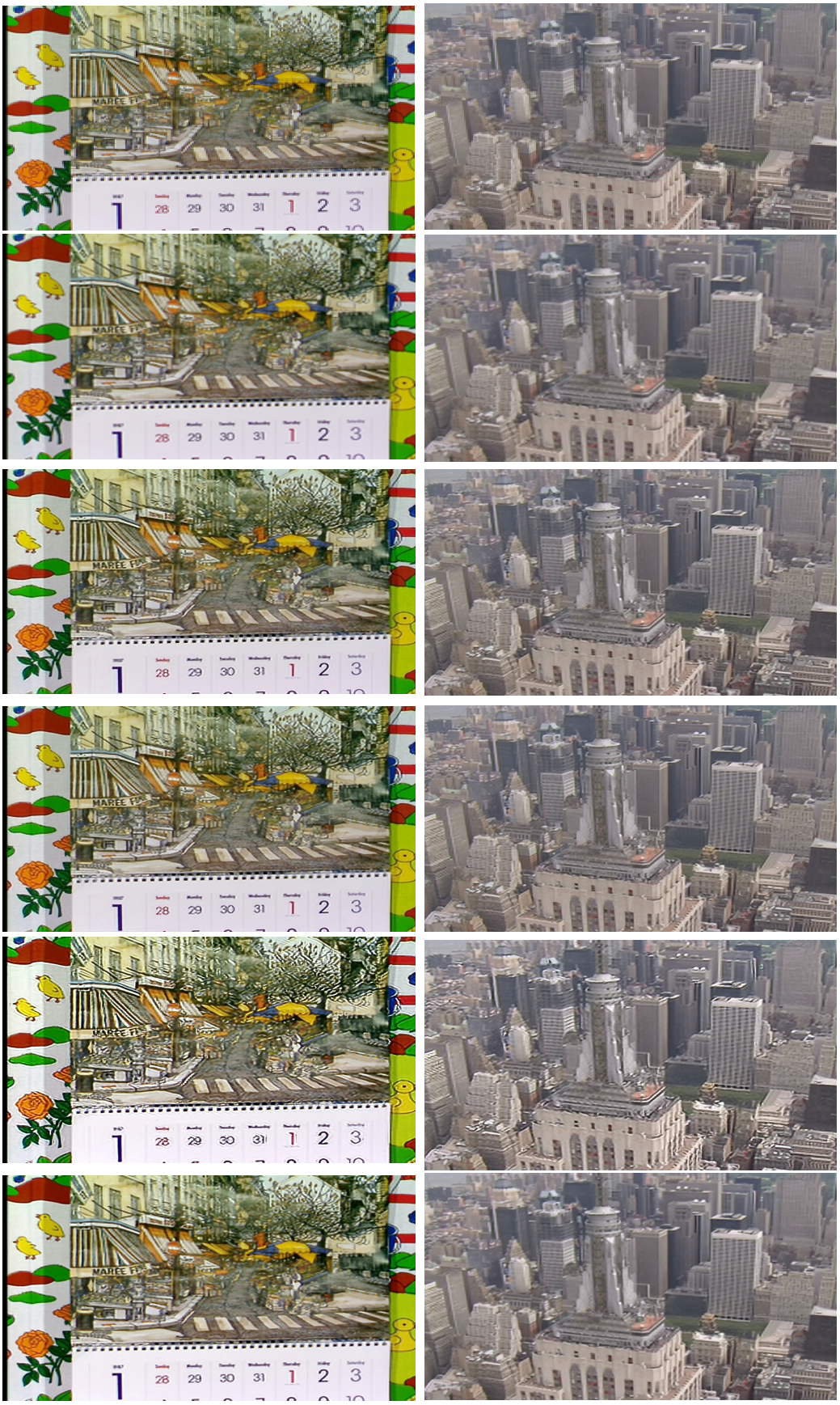

Fig. 3. Super-resolution results for a sample frame of both, the walk and foliage sequences used for testing with and upscale factor of 3. From top-to-bottom: left low-resolution input frame, bicubic upsampling [21], the results yielded by SRCNN [14], VSRNET [18], a network akin to ours trained with a mean-squared error loss and our residual-trained network.

Fig. 4. Super-resolution results for an up-scaling factor of 4 on two sample frames from the Walk and Foliage video sequences (left-hand column) and a corresponding image region detail (right-hand column). From top-to-bottom: input image frame, bicubic up-sampling [21] and the results yielded by SRCNN [14], VSRNET [18], a network akin to ours trained with a mean-squared error loss and our residual-trained network.

results with respect to the left input frame. Note that, for the video sequences used here, if we view this image as the n^{th} frame, the right one would then be that indexed $n+1$. This, in turn, implies that, for a testing sequence of N frames, we obtain $N-1$ high-resolution frames after testing the imagery under study.

B. Implementation

For purposes of coding our network, we have used *MatConvNet* [25] as the basis of our implementation and chosen standard stochastic gradient descent with momentum as the optimisation method for training, where the momentum and weight decay parameters are both set to 1. For the up-sampling step, we have used a nearest neighbour approach and used a decreasing learning rate schedule.

This schedule is as follows. At the start of the training process, we have set λ to 10^{-6}. The learning rate is then reduced to $\lambda = 10^{-7}$ after completing half of the training process. To this effect, we have used 3000 batches of 100 image pairs and trained over 1000 epochs.

All our experiments have been carried out on a workstation with an Intel i7, 3.7 GHz processor with 32GB of RAM and an Nvidia 1080Ti GPU with 11 GB of on-board memory. On our system, training took approximately 14 hrs., which is a major improvement with respect to SRCNN [14], which takes several days to train and is better than VSRNet [18] (22 hrs.), being comparable to the network analogue to ours trained using the least-squared error (12 hrs.).

C. Results

We now turn our attention to the results yielded by our network and alternatives elsewhere in the literature. It is worth noting, however, that, for comparison, and to our knowledge, there is no stereo super-resolution network analogue to ours. Therefore, we present comparison with a network with the same configuration as ours trained using the mean-squared error instead of the residual and two state of the art alternative deep networks which have been proposed for either, single-image super-resolution (SRCNN [14]) or video (VSRNET [18]) super-resolution network. This is important since the first of these, *i.e.* SRCNN [14], employs a single image at input and, therefore, does not have to deal with the paralax and registration errors introduced by the stereo baseline. VSRNET [18], in the other hand, employs 5 frames at input and, therefore, has much more information and image structure available to compute the super-resolved image. For each of the two alternatives, we have followed the authors and used their code and training schemes as reported in [14] and [18].

We commence by presenting qualitative results on the testing videos using our network and the alternatives. To this end, in Figures 3 and 4 we show the super-resolved images recovered by the methods under consideration and the input imagery. In Figure 3, we show, from top-to-bottom, the input low resolution image image used as one of the frames used as input to our method and the alternatives and the results yielded by bicubic upsampling [21], SRCNN [14], VSRNET [18], a similar network trained using the least-squared error and our approach.

In Figure 4, we show, in the left-hand and third columns, results on two sample frames of the Walk and Foliage sequences whose degraded analogue is used as a low-resolution input. The second and right-hand columns show details of the imagery. In the figure, from top-to-bottom we show the image frame in its native high-resolution before being degraded to be used for testing and the results yielded by bicubic upsampling [21], SRCNN [14], VSRNET [18], a similar network trained using the least-squared error and our approach.

From the results, note that our method, despite only using two frames at input, yields results that are comparable with those yielded by VSRNET [18]. This can be appreciated in the dates on the calendar in Figure 1 and the car on the detail in Figure 2. Moreover, the output of our method when applied to the hat of the pedestrian in Figure 2 is in better accordance with the high-resolution image frame than that delivered by the alternatives. As compared with the network trained using the mean-squared error, we can appreciate from the details in Figure 4 that our residual trained network introduces less "rigging" on the output imagery. This can be noticed in the background on the pedestrian detail and the trees next to the car.

In Table I, we compare our network, with its residual training scheme with a similar one trained using the least-squared error. In the table, we show the average peak signal

TABLE I
MEAN PSNR FOR THE VIDSET4 VIDEOS AT AN UP-SCALE FACTOR OF 4 FOR BOTH, OUR NETWORK AND AN AKIN DEEP NET TRAINED USING THE MEAN-SQUARED ERROR.

Training	City	Calendar	Walk	Foliage	Overall
Ours	24.1515	19.9039	25.9247	22.7161	23.212
Mean-squared error	23.8922	19.816	25.5949	22.6796	23.038

TABLE II
MEAN PSNR FOR OUR METHOD AND THE ALTERNATIVES WHEN APPLIED TO THE VIDSET4 VIDEOS

Dataset	Up-scale factor	Bicubic up-sampling	SRCNN	VSRNET	Ours
City	2	27.9265	-	-	29.3219
City	3	24.4695	25.6432	25.7138	24.6208
City	4	23.7672	24.2101	24.3406	24.1515
Calendar	2	22.8656	-	-	23.9863
Calendar	3	19.8479	21.4807	21.5185	20.0590
Calendar	4	19.2183	20.0731	20.0141	19.9039
Walk	2	30.4016	-	-	33.0468
Walk	3	25.0886	28.8565	28.9370	25.4551
Walk	4	24.6654	26.1918	26.2328	25.9247
Foliage	2	26.959	-	-	28.7115
Foliage	3	22.9907	24.9970	25.1470	23.2202
Foliage	4	22.2351	23.0067	23.0995	22.7161
Overall	2	27.122	-	-	28.901
Overall	3	23.108	25.351	25.439	23.356
Overall	4	22.485	23.422	23.471	23.212

to noise ratio (PSNR) over the four testing sequences for an upscale factor of 4. From the figure, we can readily appreciate the improvement in performance induced by the use of the residual to train the network.

Finally, in Table II, we show a quantitative evaluation of our method against the alternatives. To do this, we have used, again, the PSNR over each of the testing image sequences. In the table, we show the PSNR when an upscale factor of 2, 3 and 4 are used. As mentioned above, the three methods differ in terms of the number of images taken at input and, hence, the comparison presented here should be taken with caution. Note that, despite taking two input images instead of five at input, our method is comparable with VSRNET [18] with an upscale factor of 4. Our method its also competitive with respect to SRCNN [14], which is a single image method and, hence, does not have to account for the image displacement in the stereo pairs.

IV. Conclusion

In this paper, we have presented a deep convolutional network for stereo super-resolution. The network is comprised by two sub-nets that share a single output layer. Each of these nets is ten layers deep, which allows them to exploit contextual information across the image even when the filter size is 3×3. We have trained the network using the residual image. Our network is devoid of gradient clipping operations and converges faster at training than other alternatives elsewhere in the literature. We have also illustrated the utility of our network for stereo super-resolution and compared out results to those yielded by alternatives elsewhere in the literature.

References

[1] P. E. Eren, M. I. Sezan, and A. M. Tekalp, "Robust, object-based high resolution image reconstruction from low-resolution video," *IEEE Transactions on Image Processing*, vol. 6, no. 10, pp. 1446–1451, 1997.

[2] T. Bishop, S. Zanetti, and P. Favaro, "Light field superresolution," in *IEEE International Conference on Computational Photography*, 2009.

[3] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multi-frame super-resolution," *IEEE Transactions on Image Processing*, vol. 13, pp. 1327–1344, 2003.

[4] R. Y. Tsai and T. S. Huang, "Multiple frame image restoration and registration," in *Advances in Computer Vision and Image Processing*, 1984, pp. 317–339.

[5] S. P. Kim, N. K. Bose, and H. M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, no. 6, pp. 1013–1027, 1990.

[6] N. K. Bose, H. C. Kim, and H. M. Valenzuela, "Recursive implementation of total least squares algorithm for image reconstruction from noisy, undersampled multiframes," in *IEEE Conference on Acoustics, Speech and Signal Processing*, vol. 5, 1993, pp. 269–272.

[7] B. C. Tom, A. K. Katsaggelos, and N. P. Galatsanos, "Reconstruction of a high resolution image from registration and restoration of low resolution images," in *IEEE International Conference on Image Processing*, 1994, pp. 553–557.

[8] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Join map registration and high resolution image estimation using a sequence of undersampled images," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1621–1633, 1997.

[9] N. Nguyen and P. Milanfar, "An efficient wavelet-based algorithm for image super-resolution," in *IEEE International Conference on Image Processing*, vol. 2, 2000, pp. 351–354.

[10] D. C. Youla and H. Webb, "Image registration by the method of convex projections: Part 1," *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 81–94, 1982.

[11] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1167–1183, 2002.

[12] M. Protter and M. Elad, "Super resolution with probabilistic motion estimation," *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1899–1904, 2009.

[13] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Computer Vision and Pattern Recognition*, 2008.

[14] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, 2014.

[15] ——, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[16] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Computer Vision and Pattern Recognition*, 2016.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, 2014.

[18] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, pp. 109–122, 2016.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, 2015.

[20] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[21] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6.

[22] C. Liu and D. Sun, "On bayesian adaptive video super resolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 346–360, 2014.

[23] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730.

[24] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*. Wiley, 2000.

[25] A. Vedaldi and K. Lenc, "Matconvnet – convolutional neural networks for matlab," in *Proceeding of the ACM Int. Conf. on Multimedia*, 2015.