

LoGSRN: Deep Super Resolution Network for Digital Elevation Model*

Dongjoe Shin¹ and Stephen Spittle²

Abstract—The spatial resolution of a Digital Elevation Model (DEM) plays a crucial role in many practical remote sensing applications. However, it is normally limited by the spatial resolution of the raw input imagery, from which a DEM is derived. One solution to enhance the limited resolution of a DEM during the post-processing, is fusing previously obtained high resolution DEM data. This data-driven approach appears particularly promising, considering the recent success of a deep convolutional network in single image super resolution (SISR). In this paper, we propose a new SISR network that can recover a high resolution DEM. Instead of configuring a single network directly mapping low resolution depth values to high resolution depth values, we propose a new model consisting of 3 subnetworks, i.e. a) extracting feature maps; b) inferring the high frequency details; c) refining the result combining the low resolution input with the details from b). This is similar to LapSRN [1] in that both adopt a Laplacian image pyramid to model the scaling process in SISR. However, the proposed method implements a much deeper subnetworks efficiently with multiple recursive feedback and feedforward connections, and an additional Laplacian of Gaussian (LoG)-based loss function help to produce more effective training results. In this research, we also produce a high quality DEM dataset obtained from optical and lidar sensors, from satellites and aircraft respectively, covering different scenes found in remote sensing applications. Our experiments demonstrate that the proposed model performs better than other standard deep SISR models in terms of the training convergence and the Peak Signal to Noise Ratio (PSNR) of a reconstructed DEM.

I. INTRODUCTION

A digitised image generally suffers from a range of signal distortions, including the aliasing effect from a sampling process, the noise from an optical sensor, and the blur effect from the lens aberration and the sensor motion [2]. Therefore, recovering a higher resolution image from a degraded single or a sequence of low resolution images is a complex and ill-posed problem.

One obvious solution for increasing the spatial resolution is simply integrating more optical sensors, however there is a physical limitation for the pixel size to avoid the shot noise [3]. In addition, considering extreme operating conditions of a remote sensing vehicle, changing hardware specification is not always feasible. Thus, the cost of high resolution terrain models is generally high even simply for the visualisation

*This work was supported by the Researchers in Residence programme hosted by Satellite Applications Catapult, UK

¹Dongjoe Shin is with School of Creative Technologies, University of Portsmouth, Winston Churchill Avenue, Portsmouth, PO1 2DJ, UK Dongjoe.Shin@port.ac.uk

²Stephen Spittle is with Space Applications Catapult, Electron Building, Fermi Avenue, Harwell, Oxford, OX11 0QR, UK Stephen.Spittle@sa.catapult.org.uk

purpose, as they are often derived products requiring more than 2 high resolutions satellite images.

Due to the growing demand of geospatial raster products, a range of techniques has been applied to remote sensing data, since the beginning of the Super Resolution (SR) research [4]. Although there are many mission-specific or sensor-specific Multi-Image Super Resolution (MISR) solutions [5], [6], [7], [8], recent trends show that learning based SISR is more preferable due to cost and access to data [9], [10], [11], [12]. The proposed approach is in line with the learning-based SR approach. However, instead of enhancing raw satellite imagery, our interest is enhancing its derived products, i.e. DEMs. Having a sensor independent SR model is particularly useful for developing a unifying post-processing pipeline. Also, it makes easier to collect sample data, given that many publicly available DEMs are produced from various sensor types, such as synthetic aperture radar, lidar, and stereo imagery.

Since a DEM is a 2D float depth map, some SR techniques developed for image restoration can be directly applicable. However, it should be noted that the variance of height values is more significant than that of a 8-bit image. Furthermore, the linear models describing the distortion of the image digitisation is no longer valid. More interestingly the details required for a high resolution (HR) depth map are significantly different depending on scene types; e.g. many artificial objects found in an urban scene should need sharp depth transitions after rescaling, whilst the depth values in natural scenes are relatively smooth. Therefore, we need a highly non-linear mapping function that can understand the surrounding scene for recovering a HR depth map. To address this, we develop a deep SISR model optimised for the depth map enhancement. Motivated by LapSRN [1], the proposed method adopts a Laplacian image pyramid but we implement much deeper structure more efficiently using multiple skip connections, and additional LoG based loss helps to recover the sharp depth discontinuities.

To understand the details, this paper will briefly review some key deep SISR models in Sec. II. In the following section, we also explain our design idea, data augmentation, and loss functions. Some experimental results obtained during the tests are presented in Sec. IV, followed by our conclusions and future work in Sec. V.

II. RELATED WORK

Since the recent success of Convolutional Neural Networks (CNNs) in general image classification [13] and object

detection [14], it has been applied to many classic image processing applications and demonstrated superior performance over existing solutions.

One way to understanding how a CNN works is seeing it as layers of special non-linear image filters, consisting of convolutions with small kernels (e.g. 3×3) followed by thresholding using a sigmoid function with the optional use of max pooling. The clever part of a CNN approach is that we do not need to specify any filter coefficients manually. Instead, we can simply configure the network topology and let it estimate the best filter coefficients by itself from the training data. Although the training process often sounds very challenging due to the large number of unknowns and the complex network topology, it can be solved effectively with standard gradient-based optimisation methods in a parallel computing environment. Furthermore, many training techniques developed for neural networks (e.g. stochastic gradient descent, batch normalisation) can be transferred to help facilitate the process.

The first successful CNN model for SISR is SRCNN [15]. In this approach, the SR problem is posed as a non-linear image mapping, which needs to map a set of the image features from convolutional filters to the pixel values for a higher resolution image. To simulate these processes, SRCNN is configured with a 3-layer CNN for the initial feature extraction, the non-linear mapping, and final channel reduction. The authors also report that this simple structure is closely related to a sparse representation approach [16], [17] with a different non-linear mapping strategy. However, the shallow structure of SRCNN is not sufficient to address the strong depth variation observed in urban DEMs and generally produces over smooth results.

Although the number of trainable parameters in a SRCNN is small, there is still room for further optimisation for real time video processing. Fast SRCNN (FSRCNN) proposed by Dong et al. [18] argues about the initial bicubic interpolation applied to a low resolution (LR) input in SRCNN, which can slow down the feature extraction process. Instead, they directly extract small feature maps and then recover a desired spatial resolution using transposed convolutional layers later [19]. Shi et al. also point the same speed issue caused by the interpolated input image in SRCNN, but they address it differently by using sub-pixel convolution layers, which reshuffles the low resolution feature maps in a specific order to fill the desired resolution [20].

The performance of CNNs generally depends on the number of layers; in general the more (i.e. deeper) layers the better the performance. However, a deeper CNN also increases the risk of overfitting, and the increased number of unknown parameters in a deeper network makes the training process much longer and requires more training samples. One solution for configuring a deeper SR network whilst maintaining the number of trainable parameters low, is reusing the same convolutional layer recursively. This Recursive CNN (RCN) can also reflect more contextual information from a larger scale because the recursion of the same filter is effectively equivalent to applying a sequence of

filters with gradually increasing kernel size. Deeply recursive RCN (DRCN) proposed by [21] implements this idea for SISR, and resolves the training stability issue caused by a RCN with higher recursions.

Another interesting trend in deep learning over the past few years is creating a deeper network using a residual network (ResNet), which is a subnetwork constructed by an additional skip connection between every few CNN layers. He et al. demonstrate that a simple skip connection can effectively tackle the gradient vanishing/exploding problem observed in a deeper network [22]. Many recent deep SISR models adopt the ResNet idea and report the improvement of the training stability and prediction performance [23], [24]. Amongst many ResNet-based deep SR networks, the work from Ledig et al. (called SR-GAN) [25] draws particular interest from the community, as it combines a ResNet structure with an adversarial training framework. Lately, Lim et al. propose a more optimised version of the ResNet structure for SR in their Enhanced Deep Residual networks for SISR (EDSR) which won the NTIRE2017 SISR challenge [26].

Laplacian SR Networks (LapSRN) [1] are also an intriguing development in the recent deep SISR literature. Unlike the previous deep SR methods, this model is developed to deal with a large scale change using two parallel networks mimicking a Laplacian image pyramid. For example, the feature extraction branch of LapSRN is designed to produce high frequency details from feature maps, whilst the image reconstruction branch is designed to produce a scaled up image by merging them. The authors also mention that the potential of recursive layers sharing the same parameters in deep SISR networks.

When developing SISR for DEMs, we try to adopt the various features of the recently developed deep SISR architectures. We notice a strong link between LapSRN and the ResNet structure. For example, the skipping connection used in the ResNet enforces the network to learn the difference between an input and an output (i.e. high frequency image difference). Similarly, LapSRN is required to learn the high frequency image components (i.e. image difference) from the feature extraction network. Based on this observation, we adopt an interpolated skipping connection, which can combine the advantage of the ResNet and the LapSRN used in SISR. The proposed model also adopts the RCN idea to implement a deeper network for feature extraction, and an additional feedforward connection is used in the final refinement network similar to EDSR.

III. PROPOSED METHOD

A. Network configuration

The first LoGSRN model (see Fig. 1a) developed in our test is motivated by the simple SRCNN structure. In this initial model (called LoGSRN- α), we construct two subnetworks; one for feature extraction from DEMs and another for non-linear mapping from the extracted features to the height values of an enlarged DEM grid, respectively.

Compared to SRCNN, there are three additional features in LoGSRN- α . For example, the subnetwork for the feature

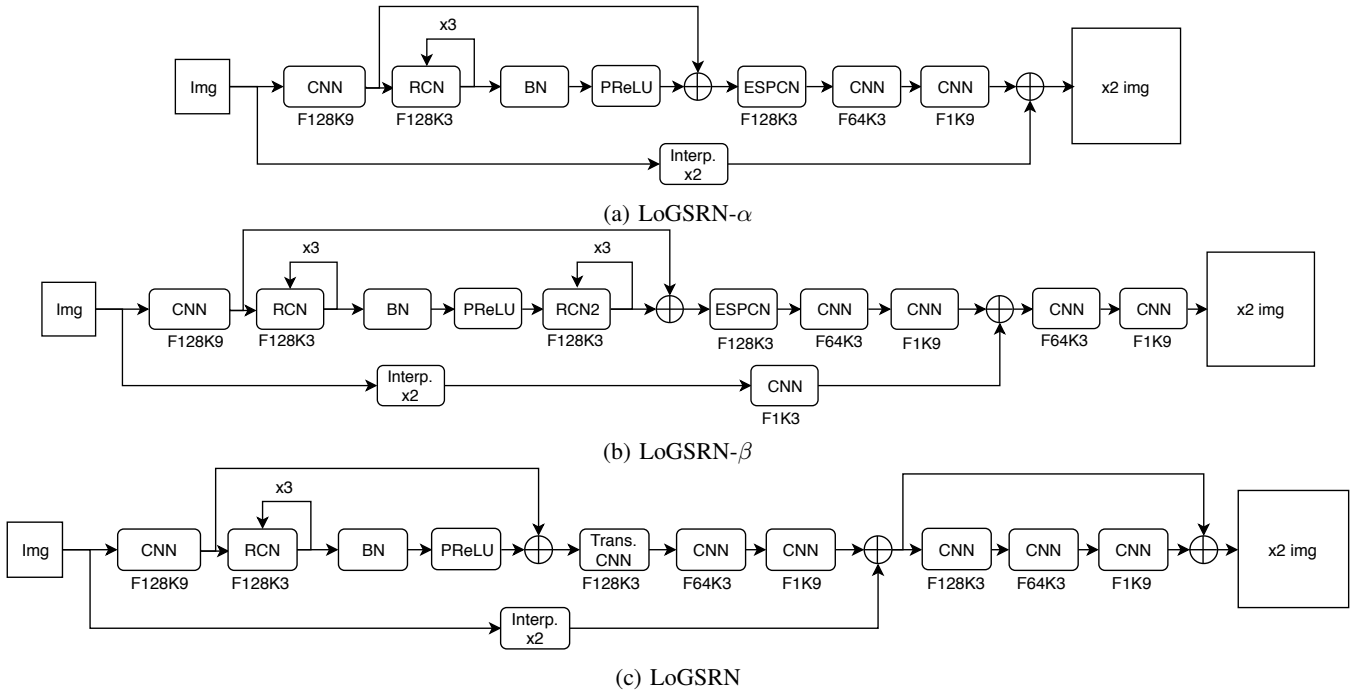


Fig. 1: Example of LoGSRN and its variants. a letter F and K under a CNN block denote the number of convolutional layers and the size of a filter kernel, respectively: (a) an alpha version of LoGSRN employing an interpolated skip connection and a recursive network; (b) a beta version of LoGSRN including additional RCN unit motivated by a standard ResNet structure and more refinement CNNs; (c) Final version of LoGSRN model consisting of 3 subnetworks for extracting feature maps, inferring LoG for a HR depth map, and final refinement with a skip connection.

extraction (i.e. up to the first feedforward connection point in Fig. 1a) has a RCN layer to implement a deeper feature extraction network without incurring too many unknown parameters. The two feedforward connections (i.e. the interpolated skip connection in the outer loop and the standard skip connection in the feature extraction) are used to improve the training stability and enforce the network to learn the high frequency components implicitly. A batch normalisation (BN) and a parametric rectification linear unit (PReLU) are also added to produce sharper depth transition when estimating feature maps. The last subnet of LoGSRN- α (i.e. the processing blocks from the Efficient SubPixel Convolutional Network (ESPCN) [20] to the last CNN block in Fig. 1a) is scaling the extracted feature maps before merging them with an interpolated depth map.

To maximise the benefit of the increased depth generated by the RCN, the beta version of LoGSRN (see Fig. 1b) includes a post-RCN unit in the feature extraction (see the second RCN unit after the PReLU block in Fig. 1b). This design change resembles a standard ResNet structure which is constructed by a sequence of CNN-BN-ReLU-CNN [22]. The additional CNN in the interpolated skip connection makes LoGSRN- β closer to the image reconstruction branch of LapSRN, and this could be more beneficial particularly when there is stronger signal distortion in a LR depth map.

In the final version (see Fig. 1c), we develop a model containing 3 subnetworks. The purpose of each subnet can be summarised as a) extracting feature maps, b) inferring

the high frequency details (i.e. the required details for a HR depth map), and c) final refinement, respectively. The purpose of the first two subnets (i.e. up to the third CNN block in Fig. 1c) is in essence translating a LR depth map to HR LoG values. The second subnet (i.e. from the transposed CNN block to the third CNN block in Fig. 1c) is the same as the last subnet of the previous versions except that the ESPCN is now replaced with a transposed convolutional layer. This change has been implemented as we found that ESPCN is prone to producing the block artefacts from the shuffling of the feature maps from the zero-padded interpolation (see the first two SRCNN results in Fig. 2). We also attach a final ResNet block which will be used to refine the merged depth map combining the interpolated LR depth map with HR LoG estimation.

B. Loss functions

For a loss function for LoGSRN- α and LoGSRN- β , we use a standard Mean Square Error (MSE) of the depth difference between the estimation and the true depth value, i.e.

$$l_{depth}(\mathcal{X}, \mathcal{Y}; \vec{\theta}) = \frac{1}{|\mathcal{X}|} \sum_{\vec{x}_i \in \mathcal{X}} |\vec{y}_i - f(\vec{x}_i; \vec{\theta})|^2, \quad (1)$$

where \mathcal{X} and \mathcal{Y} respectively represent a set of LR depth maps and the corresponding HR depth maps in a mini batch, and $f(\vec{x}_i; \vec{\theta})$ is a prediction function that produces an estimated HR depth map from a LR depth map \vec{x}_i with the trainable

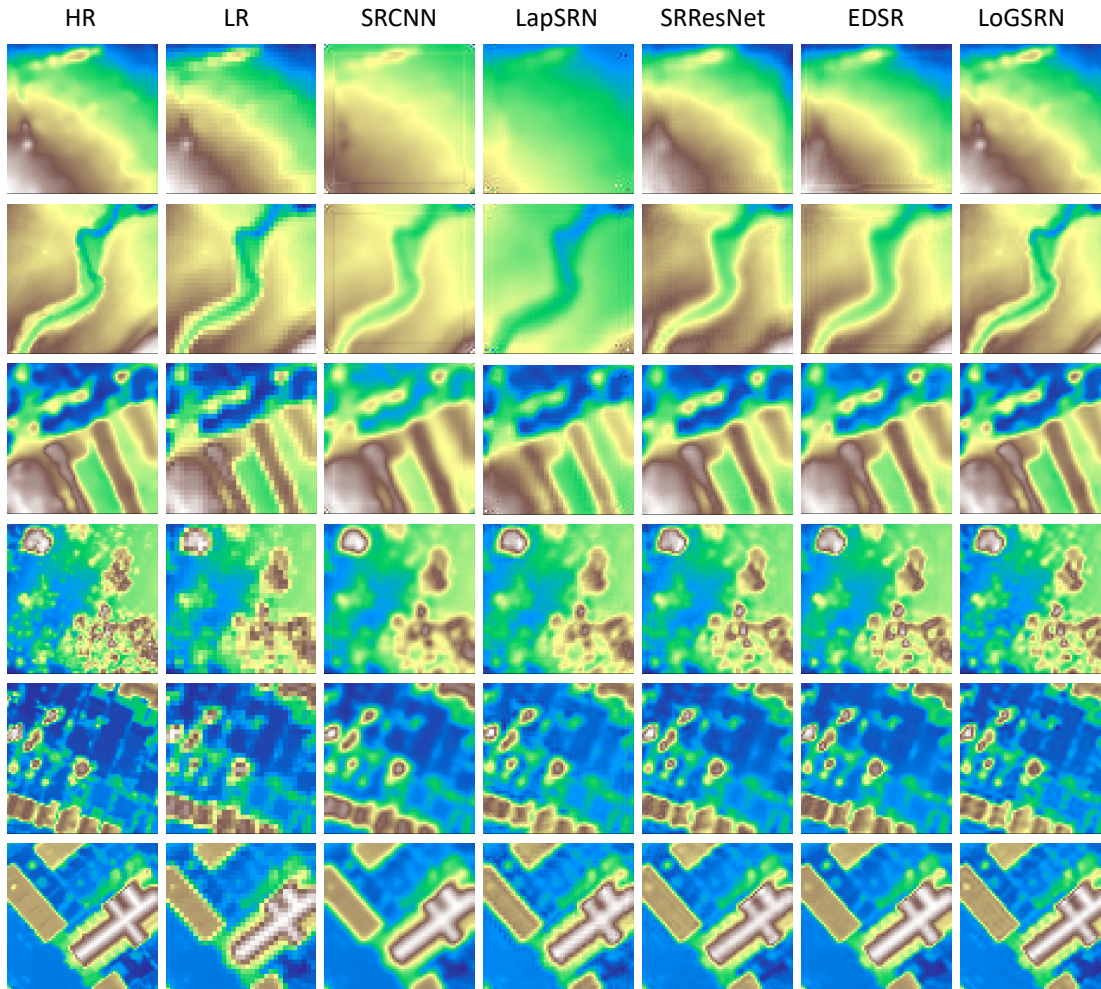


Fig. 2: Example of deep SR reconstruction results for natural, suburban, and urban scenes

network parameters $\vec{\theta}$, i.e. ideally a well trained model should be able to produce $f(\vec{x}_i; \theta) \simeq \vec{y}_i$.

The above MSE loss generally works well and produces high PSNR. However, it is not sufficient at times capturing the abrupt depth changes in many artificial structures in a urban scene. To improve the sharpness of a scaled depth map further, we add an additional LoG based loss function in the final model, such that

$$l_{LoG}(\mathcal{X}, \mathcal{Y}; \vec{\theta}) = \frac{1}{|\mathcal{X}|} \sum_{\vec{x}_i \in \mathcal{X}} |\nabla \vec{y}_i - g(\vec{x}_i; \vec{\theta})|^2. \quad (2)$$

The function $g(\cdot)$ in (2) represents the high frequency details estimated by the proposed LoGSRN (i.e. the result before merging with the interpolated skip connection in Fig. 1c). Thus, what (2) does is explicitly comparing the estimated high frequency details against the LoG of y_i (i.e. $\nabla \vec{y}_i$). This makes the first two subnets of the proposed LoGSRN simulate the scaling process of a Laplacian pyramid explicitly.

The combined loss for LoGSRN is defined as the weighted sum of two losses. i.e.

$$l(\mathcal{X}, \mathcal{Y}; \vec{\theta}) = \lambda l_{depth}(\mathcal{X}, \mathcal{Y}; \vec{\theta}) + (1 - \lambda) l_{LoG}(\mathcal{X}, \mathcal{Y}; \vec{\theta}), \quad (3)$$

where λ is a weighting coefficient between 0 and 1, and $\lambda = 0.95$ was used in our experiments. We also approximate $\nabla \vec{y}_i$ in (2) as the Difference of Gaussian (DoG) to simplify the computation.

C. Depth DB and data augmentation

To train and validate the proposed models, we construct a new depth DB. Raw DEMs produced from different missions and different sensors are collected to increase the variation within the training samples.

The spatial resolution of the raw DEMs ranges from 0.25m to 1m, and captures various topological features from natural (e.g. river, field, mountain - see the HR images of the first two rows in Fig. 2) and urban scenes (e.g. road, building - see the HR images of the last two rows in Fig. 2). To populate samples from urban scenes, we used DEMs from London (0.5m, lidar), and a subset of Didcot (1m, lidar) and Bath (0.25m, optical sensor) in the UK, whilst DEMs from Yorkshire (1m, lidar) and Region Four in Chile (1m, satellite stereo imagery) are collected for covering natural scene. The rest of the Bath and the Didcot data defines suburban scenes, which are similar to natural scenes but contain sparse artificial structures.

Since the proposed system is designed to return a 64×64 depth map from a 32×32 depth map, we initially subdivide each raw DEM into 64×64 depth patches and apply downsampling to populate smaller training samples. A small Gaussian blur is added dynamically during the training to improve the generalisation of the trained model, and the higher resolution raw DEMs from the Bath and London are normalised to 1m resolution using a simple sampling process before the image cropping process. The resulting depth patches are also filtered during the preprocessing, so that we can remove the patches containing too many unknown values from the previous DEM production process. Raw DEMs with strong visible artefacts (e.g. the stitching pattern from the ortho-rectification or the mosaicking process) are also excluded. In the final DB, we have collected 119,816 training patches and 1,210 testing patches¹.

IV. EXPERIMENTS

To evaluate the performance of the various SISR models, this section presents the prediction results and the training results. The prediction results are obtained from the test dataset which has never been seen to the model during the training, so that they are unbiased for evaluating the generality of the trained models. On the other hand, the training results are obtained from a larger training dataset to demonstrate the training stability over the iterations, so that it helps us to understand training behaviours, e.g. how quickly a model can converge, and how noisy the training process can be.

A. Prediction results

For visual comparison, we present some predicted results from five different deep SR methods in Fig. 2. The first and the second column of the figure respectively represent HR depth patches (i.e. ground truth) and the corresponding LR patches (i.e. input for each method), which is enlarged to 64×64 for visualisation purpose. The images from column three to seven show the results produced by SRCNN, LapSRN, SRResNet, EDSR, and LoGSRN. In our test, we use a standard EDSR and SRResNet model having sixteen and five ResNet units, respectively.

Despite of the smallest number of trainable parameters, SRCNN (see the third column in Fig. 2) can perform better than LapSRN (see the fourth column in Fig. 2) particularly when dealing with the smooth depth transition observed in most natural scenes (e.g. the first two rows in Fig. 2).

However, we found that LapSRN can reconstruct the sharper depth transition in the urban scenes better (e.g. the last two rows in Fig. 2). EDSR and SRResNet show similar performance in general but the EDSR results often look slightly better for capturing the high frequency components in urban and suburban scenes. The proposed method performs best in both natural and urban scenes (see the last

¹The raw satellite DEMs used in the creation of the training and testing depth patches are derived from the satellite imagery (from DigitalGlobe Products, Imagery 2016 DigitalGlobe, Inc., a Maxar company), and the lidar data from UK Environment Agency (copyright and/or database right 2015. All right reserved)

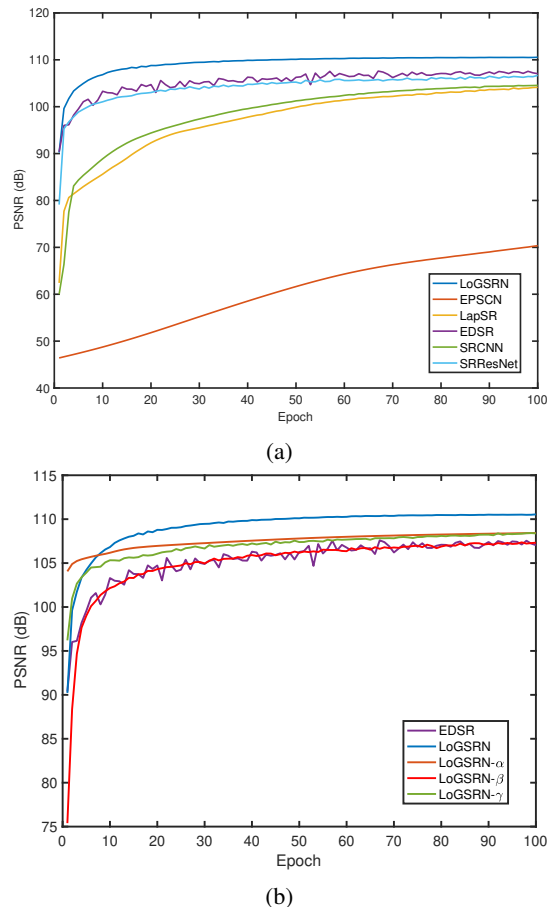


Fig. 3: PSNR with respect to a training epoch: (a) PSNR comparison with other deep SISR methods; (b) PSNR comparison with LoGSRN variants (nb. the result from EDSR is presented as a baseline)

column in Fig. 2). The results from the other LoGSRN variants and ESPCN are not included due to the page limit but the PSNR values of all methods are found in Fig. 3.

B. Training results

In our experiment, we used an Adam optimiser with a fixed learning rate (i.e. $1e^{-4}$). 32 LR depth patches are randomly selected in each batch. Since the MSE loss (1) is closely related to PSNR, all models manage to produce good PSNR convergence (see Fig. 3a) within 100 epochs. In our test, the worst PSNR was recorded by ESPCN which produces about 70dB after 100 epochs (nb. the potential max value for a 16-bit depth map is 65535, so that our PSNR is generally higher than the PSNR observed in a 8-bit image SR problem).

As reported in [26], EDSR performs better than SRResNet (which is a SR-GAN without the adversarial learning part) but tends to be more noisy over the time. LapSRN and SRCNN show similar performance although LapSRN starts with higher PSNR in the beginning. The best convergence performance was recorded by the proposed LoGSRN which can produce about 111dB.

Fig. 3b compares the convergence performance between

other LoGSRN variants. In this test, we add an additional model called LoGSRN- γ , which has the same architecture as LoGSRN but only uses the MSE loss shown in (1) to demonstrate the impact of the additional LoG loss. Interestingly, the performance of LoGSRN- β is similar to EDSR but it shows more stable convergence behaviour. LoGSRN- α performs similar to LoGSRN- γ despite of the relatively simpler structure, but the best result was achieved by LoGSRN.

V. CONCLUSIONS AND FUTURE WORK

This paper presents a new SISR network which can be used to enhance the spatial resolution of depth maps obtained from various satellite missions. The proposed model (called LoGSRN) is developed to simulate the scaling process of a Laplacian image pyramid. This is similar to LapSRN but the proposed model implements deeper network using multiple feedback and feedforward skip connections. Our test results demonstrate that the interpolated skip connection and additional LoG loss can help to achieve better PSNR, and the shared weights from the recursive network reduce the training time for a deeper network.

Although the proposed LoGSRN shows good performance with a test dataset, we have identified some limitations and future work for the proposed model. One obvious next challenge is producing a higher scale depth map beyond $\times 4$. One solution could be to append the LoGSRN sequentially as suggested in LapSRN. However, this does not utilise the similarity between scaling CNNs (i.e. the weights for the $\times 2$ CNN are closely related to that for the higher scaling CNNs). Therefore, it is worth investigating an efficient network architecture that can combine all the intermediate LR results adaptively and feedforward it to the next CNN layers. Another important question that arises from practical applications is increasing the size of the input. In theory, the proposed pipeline can ingest a larger SL depth patch, because it is a CNN model. However, the visual performance would not be better if the model is trained only with images at specific resolution. For the multi-resolution scaling, we think the hourglass structure used in the image segmentation could be useful.

REFERENCES

- [1] W. Lai, J. Huang, N. Ahuja, and M. Yang, "Fast and accurate image super-resolution with deep Laplacian pyramid networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.
- [2] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.
- [3] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–36, 2003.
- [4] D. Yang, Z. Li, Y. Xia, and Z. Chen, "Remote sensing image super-resolution: Challenges and approaches," in *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 196–200, 2015.
- [5] R. Y. Tsai and T. S. Huang, "Multipleframe image restoration and registration," in *Advances in computer vision and Image Processing*, vol. 1, pp. 317–339, CT:JAI Press Inc, 1984.
- [6] J. Yang and T. Huang, *Super-Resolution Imaging*. CRC Press, 1 ed., 2010.
- [7] B. C. Tom and A. K. Katsaggelos, "Reconstruction of a high-resolution image by simultaneous registration, restoration, and interpolation of low-resolution images," in *Proceedings, International Conference on Image Processing*, vol. 2, pp. 539–542, 1995.
- [8] D. Capel and A. Zisserman, "Computer vision applied to super resolution," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 75–86, 2003.
- [9] S. Gou, S. Liu, S. Yang, and L. Jiao, "Remote sensing image super-resolution reconstruction based on nonlocal pairwise dictionaries and double regularization," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 12, pp. 4784–4792, 2014.
- [10] B. Hou, K. Zhou, and L. Jiao, "Adaptive super-resolution for remote sensing images based on sparse representation with global joint dictionary model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2312–2327, 2018.
- [11] F. Li, X. Jia, D. Fraser, and A. Lambert, "Super resolution for remote sensing images based on a universal hidden markov tree model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 3, pp. 1270–1278, 2010.
- [12] W. Yang, X. Zhang, Y. Tian, W. Wang, and J. Xue, "Deep learning for single image super-resolution: A brief review," *CoRR*, vol. abs/1808.03344, 2018.
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.
- [14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.
- [15] C. Dong, C. C. Loy, K. He, and X. Tang, "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. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [17] M. Aharon, M. Elad, and A. Bruckstein, "k-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [18] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *The European Conference on Computer Vision (ECCV)*, pp. 391–407, Springer International Publishing, 2016.
- [19] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *The European Conference on Computer Vision (ECCV)*, pp. 818–833, Springer International Publishing, 2014.
- [20] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874–1883, 2016.
- [21] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1637–1645, 2016.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
- [23] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2790–2798, 2017.
- [24] J. Xu, Y. Zhao, Y. Dong, and H. Bai, "Fast and accurate image super-resolution using a combined loss," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1093–1099, July 2017.
- [25] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 105–114, 2017.
- [26] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1132–1140, 2017.