AFN: Attentional Feedback Network based 3D Terrain Super-Resolution

Ashish Kubade, Diptiben Patel, Avinash Sharma, and K. S. Rajan

International Institute of Information Technology, Hyderabad, India.
{ashish.kubade, dipti.patel}@research.iiit.ac.in,
{asharma, rajan}@iiit.ac.in

Abstract. Terrain, representing features of an earth surface, plays a crucial role in many applications such as simulations, route planning, analysis of surface dynamics, computer graphics-based games, entertainment, films, to name a few. With recent advancements in digital technology, these applications demand the presence of high resolution details in the terrain. In this paper, we propose a novel fully convolutional neural network based super-resolution architecture to increase the resolution of low-resolution Digital Elevation Model (LRDEM) with the help of information extracted from the corresponding aerial image as a complementary modality. We perform the super-resolution of LRDEM using an attention based feedback mechanism named 'Attentional Feedback Network' (AFN), which selectively fuses the information from LRDEM and aerial image to enhance and infuse the high-frequency features and to produce the terrain realistically . We compare the proposed architecture with existing state-of-the-art DEM super-resolution methods and show that the proposed architecture outperforms enhancing the resolution of input LRDEM accurately and in a realistic manner.

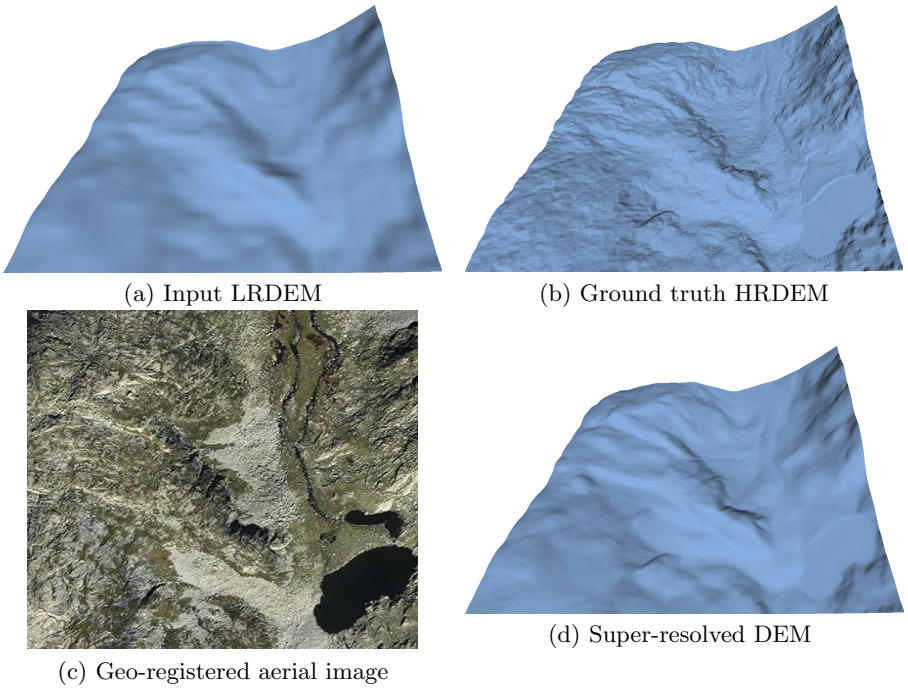
1 Introduction

Real-world terrain is a complex structure consisting of bare land, high range mountains, river paths, arcs, canyons and many more. The terrains and their surface geology are digitally represented using Digital Elevation Models (DEM) or volumetric models. The terrain data coupled with Geographical Information Systems (GIS) extract topological information for various applications including modeling water flow or mass movements, analyse the dynamic behaviour of the earth surface, perform disaster mitigation planning such as flood modeling, landslides, etc. Real-time simulations of terrains are used for fast adaptation and route planning of aerial vehicles such as drones, aircrafts and helicopters, to name a few. Realistic terrain rendering also finds its application in ranging simulations, entertainment, gaming, and many more. As the visual detail and depth in many of these applications, mentioned above, demand terrain information of high resolution and fidelity, capturing or generating such information, as accurately as possible, is the need of the hour.

Diversity and combinations of the complex topological structures make capture/synthesis and analysis of the terrain a challenging task while taking realism

into consideration. For instance, computer games with high realistic graphic environments include terrain features for users to experience better realism and allow for detailed exploration. The synthetic or amplified terrain can be used as a background for science fantasy films as well, as the synthetic terrain does not exist and amplified terrain may be difficult for the filming process.

However, DEMs captured with recent remote sensing sensors are still of relatively low-resolution (> 2 meters per pixel) and very few geographical locations are captured in high-resolution using airborne LiDAR technology due to high processing requirements. An alternate solution to this problem is to transform the captured low-resolution DEMs (LRDEM) to super-resolved DEMs termed as terrain modeling in general. Existing terrain modeling process can be broadly classified as terrain amplification and terrain synthesis. Terrain amplification enhances the high frequency 3D texture details of the scanned low resolution terrain captured from the real world, thereby making it as close as possible to actual ground truth terrain. On the other hand, terrain synthesis deals with generation of terrain with specific user controls giving a near-realistic appearance.



(a) Input LRDEM (b) Ground truth HRDEM

(c) Geo-registered aerial image (d) Super-resolved DEM

Fig. 1. Views of the terrain at different resolutions and corresponding aerial image.

Our primary focus in this work is on terrain amplification of LRDEM (Fig. 1(a)) with aim to obtain super-resolved DEM (Fig. 1(d)) terrain models with high fidelity to the ground-truth (Fig. 1(b)) terrain structures.

Some of the earliest methods for terrain amplification employed dictionary of exemplars to synthesis high resolution terrains [1, 2], while some other efforts

in the literature used erosion simulations to mimic the terrain degradation effects [3, 4]. Owing to recent advancements in deep learning literature for super-resolution of real world RGB images [5–10], some recent efforts have adopted these ideas for DEM super-resolution. DEM Super Resolution with Feedback Block (DSRFB) [11] is one such method that attempts to incrementally add high frequency terrain details to the LRDEM in high dimensional feature space using deep learning framework. Another line of work attempted to exploit the terrain information from alternate modalities like aerial (RGB) images (Fig. 1(c)) that are geo-registered with low resolution DEMs by performing fusion in feature space, e.g., Fully Convolutional Networks (FCN) proposed in [12]. However, despite using RGB information in DEM super-resolution task, such methods perform poorly in cases of land regions covered with dense vegetation or heavy snowfall. On the other hand, by not availing such modalities (like in DSRFB), we may refrain from exploiting the complementary information captured by RGB images primarily for bare terrain.

In this paper, we aim to utilize these complementary modalities in a more efficient and effective manner using the concept of selective fusion in feature space. Attention networks, applied to applications like image captioning [13], allow such selective fusion in deep learning framework. Therefore, we aim to design an integrated attention module that enables learning of selective information fusion from multiple modalities. In our setup, where we have two modalities viz aerial image and DEM, we use attention mechanism to selectively pick high frequency details from one modality and discard from the other. Our joint attentional module generates attention mask, which serves as a weight factor deciding the contribution of each modality.

Thus, we propose a novel terrain amplification method for the DEM representation of real world terrains. We propose supervised learning based fully convolutional neural network (CNN) with LRDEM and corresponding high resolution aerial image as an input and Super-resolved DEM as an output. The architecture of the CNN constitutes a feedback neural network with attention mechanism where the attention mask itself is also allowed to refine its response over the iterations. The high frequency details are added to the DEM using the features extracted from the corresponding high resolution aerial image using the Feature Extraction module. In order to capture high frequency details, we minimize the L1 loss. The overall architecture of the proposed Attentional Feedback Network (AFN) is shown in Fig. 2.

We compare the performance of the proposed methods with other state-of-the-art super-resolution methods for DEM in a quantitative and qualitative manner and are able to achieve better performance in terms of reduced number of parameters as well as inference time. More precisely, proposed AFN solution shares the parameters across feedback loop for incremental fusion in feature space with just 7M parameters whereas other SOTA architectures like [12] use an order of 20M parameters. Being leaner model, it achieves better performance 50% faster than the average inference time required by [12] on similar hardware.

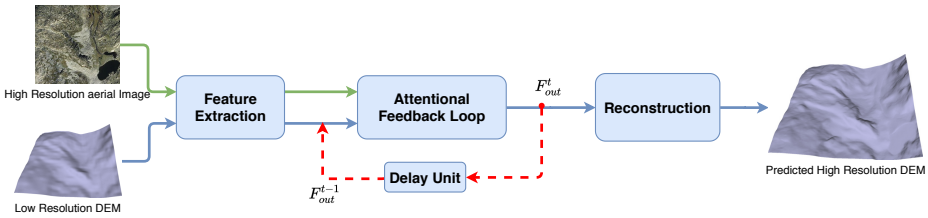


Fig. 2. Proposed Attentional Feedback Network Architecture

2 Related Work

Generating high-resolution of DEM from a low-resolution DEM can be thought of as enhancing or adding the high frequency details like texture patterns, sharp edges, and small curves which often are lost or absent in the low-resolution DEM. With recent success of deep learning, super-resolution of natural RGB images has achieved state-of-the-art performance. However, very few attempts have been made to apply super-resolution to enhance the resolution of DEMs. The possible reasons for fewer attempts could be difference of underlying features, size of features, different textures, and salient objects. Earlier attempts by [12] have explored the new paths to apply super-resolution to DEMs and successfully demonstrated that deep learning solutions can be adapted to DEMs as well. To understand the challenges in this cross domain task, we would like to highlight some of the major works in respective domains in detail. This section presents a focused overview of terrain modeling, super-resolution methods for images in general and deep learning based feedback network as individual components used in computer vision community.

2.1 Terrain Modeling

Based on the underlying process acquired for terrain modeling, it is classified into three categories: procedural generation methods, physically-based simulation methods, and example-based methods. Procedural generation methods consist of algorithms that use the intrinsic properties of a terrain from the observation of the real world. Physically-based simulation methods execute computer simulations of a geomorphological process that modifies physical properties and surface aspects of a terrain. Example-based methods extract the information from scanned heightfield real world terrains and combine these information for the generation or amplification purpose. The detailed review of existing terrain modeling processes can be referred from [14].

Procedural generation methods use self repeating fractal patterns to mimic the self repeating property of a real world terrain at different scales. Perlin et al. [15] proposed the use of generating such fractal patterns for terrain modeling. By using combinations of octaves of noises and thereby creating various scales of noise and smoothness, [3] offers variations in the fractal dimensions. Analogous to mountains, rivers can also be modeled with procedural modeling and incorporated into the landscape [16]. User interaction is involved in terrain modeling

using painting and brushing on gray-scale images as the fractal's basis functions for editing in [17]. Primitive features in the form of silhouette and shadows, vector based features in the form of ridge lines, riverbeds, cliffs have been used to generate the terrain in [18] and [19], respectively. Hierarchical combination of the primitives such as riverbed, cliffs, hills is used as a tree objects in [20]. However, terrains generated using procedural methods lack the effect of natural phenomenon like erosion in their appearances. Hence, a terrain generated by procedural methods is often combined with simulation operations.

Simulation based methods use physical processes such as diffusion, erosion, temperature aided contraction, expansion, hydrological factors aided smoothening, and wind aided gradual abrasion to generate more realistic terrain. [3] presented hydraulic and thermal erosion and combined with ecosystems such as vegetation modeling. However, the heightfield is unable to represent the arches and caves present in the terrain as heightfield can represent only topmost surface in a terrain. [21] introduced layered representation for such structures with multiple layers. These structural representations have also enabled stacking multiple layers for effects of various physical and biological phenomenon. One such integration has been represented by [22], where they fused the interaction between the growing vegetation and terrain erosion by representing them into different layers.

Example-based methods are data-driven methods utilizing the information available in scanned data of real-world terrain. Sample terrain is transformed to desired terrain using user defined sketch in [4]. Patch based terrain synthesis by using a dictionary of exemplars is performed in [2, 1]. With recently successful deep learning based Generative Adversarial Networks (GANs), [23] used Conditional GANs to translate a sample terrain using interactive user sketch.

2.2 Super-resolution of Images

Different interpolations from neighbourhood information such as linear, bilinear or bicubic are trivial solutions for super-resolution of an image. However, interpolation without high frequency information leads to average out the sharp edges resulting in blur image. Sharp edges and high frequency textures are preserved using Edge Directed Interpolation suggested in [24]. Alternatively, patch based solutions [25–27] reconstruct high-resolution patches using a learned mapping between LR and HR patches. While learning the mapping between LR and HR patches, patch consistency is a major issue with patch based approaches. In order to avoid patch inconsistency, mapping between LR and HR images is learned considering an image as a single patch and extracting hand-crafted features using convolutional operators [28], gradient profile prior [29, 30], Kernel Ridge Regressions (KRR) [31].

Super-resolution task using deep learning is attempted in [32, 33] to learn the mapping between LR and HR. With ResNet overcoming the vanishing gradient problem by using skips connections in deeper networks, super-resolution of images using residual blocks is achieved by DRCN [6], SRResNet [34], Residual of Residual (RoR) [35], Residual Dense Network (RDN) [7], to name a few. With

an emerging interest in generative adversarial networks, super-resolution of an image is attempted by [8, 35]. While the trend was to go deeper apathetic to the number of parameters, DRRN [36] formulated a recursive structure to fuse features across all depths.

2.3 DEM super-resolution with Neural Networks

Though RDN [7], DRRN [36] were able to effectively utilize the low level features, the flow of information was only in forward direction, i.e., from initial layers to deeper layers. The low level features are used repeatedly, limiting the reconstruction ability of lower features in the super-resolution task of the network.

SRFBN (Super-Resolution Feedback Network) [10] was proposed to tackle this problem. SRFBN used a feedback mechanism adapting from Feedback Networks[37] in their architecture. Using a feedback mechanism has another advantage with respect to size (number of parameters) of the model. Using a recurrent structure and thereby reusing the parameters has been one of the major techniques in deep learning. Recurrent structures also helps realizing a feedback mechanism easily as recurrent structure can save states of a layer which helps in implementing the feedback component. This approach of super-resolution has been effectively utilized in [11] for DEM super-resolution task. [11] have also suggested using overlapped prediction to remove artifacts observed at patch boundaries due to discontinued structures. Even though performing comparable with then state-of-the-art, [11] network can not avail any additionally available modalities, and hence performance of their method is limited to information cues available in low-resolution DEM only. A Method based on fully convolutional networks by [12] (referred as FCN, here onwards) extract complementary information from aerial images. However, in their feed-forward setup, there is no control over features learned by initial layers of network. Also, it has been shown that FCN could perform poorly in case of unexposed land regions covered with dense vegetation or areas with heavy snowfall. This motivates us to explore solutions that enable selective extraction of features from aerial images while focusing more on learning of initial layers of the network. We propose the use of attention mechanism for adaptive utilization of features selected respectively from aerial images and DEM. Integrating attention mechanism with feedback network enables the proposed network to learn more refined lower level features.

3 Method

Despite using RGB information in super-resolution task, FCN [12] performs poorly in cases of dense vegetation or heavy snowfall. However, by not availing such modalities, like in DSRFB [11], we may refrain ourselves from improvements in super-resolution systems. We utilize these additional modalities in complementary fashion. Inspired from attentional networks applied to applications like

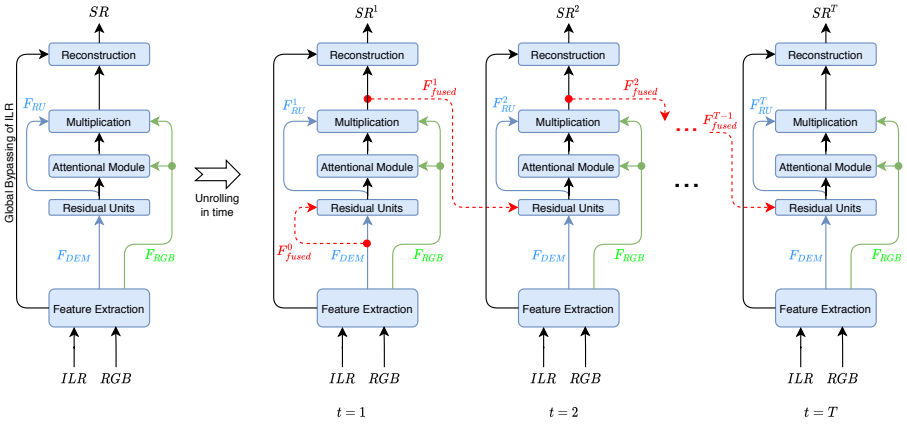


Fig. 3. Unrolled Model Structure

image captioning [13], we design a module that lets system learn to focus and extract selective information. In our setup, where we have two modalities viz aerial image and DEM, we use attention mechanism to selectively pick high frequency details from one modality and discard from the other. Our joint attentional module generates attention mask, which serves as a weight factor deciding the contribution of each modality.

Moreover, our interest is in recovering the lower level details (alternately 'high frequency' details) as edges, texture, sharp changes, etc. In a typical Convolutional Neural Network (CNN), these features are captured by the initial layers of the network. To refine the features captured by the shallow layers, we design our attention network in recursive fashion and introduce part of deep features as input back to the shallow layers. This also enables our attention mask to get updated with each time step. Thus, our network becomes a feedback network enabled with attention, we call it as 'Attentional Feedback Network' (AFN). The implementation of the feedback module is based on an RNN with T states. With each state, our model refines the lower level features learned by initial layers and enables the reconstruction of SR at each time step. The overall network architecture, once unrolled over time, has the structure as shown in Fig. 3. In next section, we explain the architectural details of each component.

3.1 Proposed Attentional Feedback Network Architecture

As shown in Fig. 3, unfolded network across time comprises of three components: A Feature Extraction Module, Attentional Feedback Module(AFM) and Reconstruction Block. We also introduce following notations used throughout this paper.

- m denotes the base number of filters
- $Conv(m, k)$ denotes a convolutional layer with output number of channels m and kernel size k

– T denotes the number of steps in feedback loop

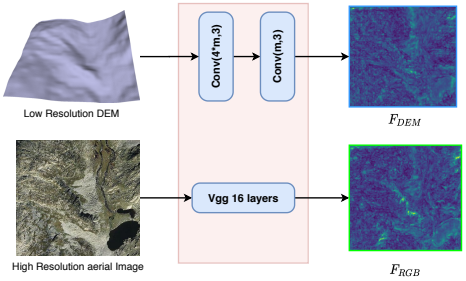
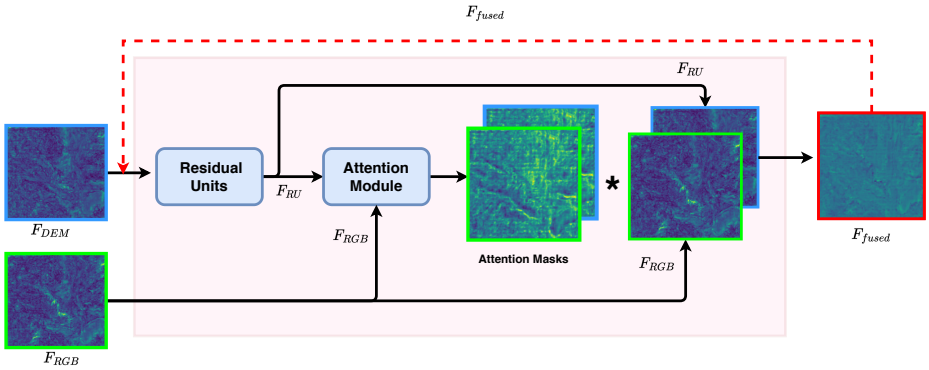
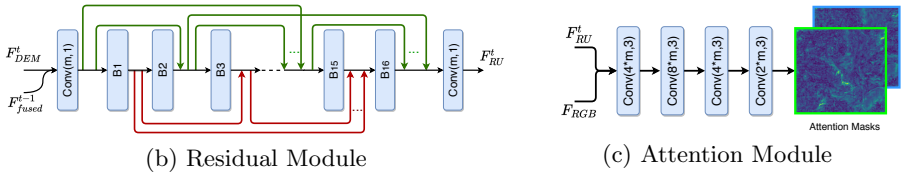


Fig. 4. Feature Extraction Module



(a) Feedback Module



(b) Residual Module (c) Attention Module

Fig. 5. Attentional Feedback Module

Input to the **Feature Extraction Module**(FE) is a pair of geo-registered LRDEM and aerial image. As shown in Fig. 4, the FE module consists of two branches of layers. Input to the first branch is LRDEM. It comprises of two convolutional layers as $Conv(4*m, 3)$ and $Conv(m, 3)$. The output of this branch is denoted by F_{DEM} (shown with blue outline). Second branch operates on aerial image. We use first two layers from pre-trained VGG-16 network [38] on Imagenet dataset to extract aerial image features. To reduce the domain shift from the aerial images to the images from Imagenet data, we fine-tune these VGG layers during training. The choice of layers has been done empirically by comparing the feature responses of the layers. First two layers are sufficient to extract most of the high frequency details. We denote the output of VGG layers as F_{RGB} (shown with green outline).

We feed the F_{DEM} and F_{RGB} to **Attentional Feedback Module**(AFM) which is the heart of our algorithm. As shown in Fig. 5, AFM consists of two sub-modules: A stack of residual units and an attention module.

Each residual unit consists of a $Conv(m, 1)$ followed by a $Conv(m, 3)$. The $Conv(m, 1)$ allows the residual unit to adaptively fuse the information from previous residual units and $Conv(m, 3)$ layer produces new m channel features to be passed towards following residual units. The residual units are denoted with B_i, where $i \in \{1, N\}$, N being an even number. As implemented by [11], we use two sets of skip connections to combine the features from residual blocks. The skip connections from $B1$ bypass the information to $\{B2, B4, B6, B8, \ldots, B_N\}$, from $B2$ to $\{B3, B5, B7, \ldots, B_{N-1}\}$, from $B3$ to $\{B4, B6, B8, \ldots, B_N\}$ and so on. Being inside the iterative feedback module, at each time step t, residual module receives a concatenated feature map of F_{DEM} and F^{t-1}_{fused}. This timely varying part F^{t-1}_{fused}, constitutes the feedback component of our network that we save at time step $t-1$ and is shown as red dashed line in Fig. 5(a). A $Conv(m, 1)$ layer has been used to compress F_{DEM} and F^{t-1}_{fused} before passing them to B1 at time step t. At current iteration, t, the outputs from units $\{B2, B4, ..., B_N\}$ are compressed by another $Conv(m, 1)$ layer to generate the output of residual module viz F^t_{RU}.

At each time step, t, the resultant output of residual module, denoted as F^t_{RU}, along with the features from the RGB branch i.e. F_{RGB}, are fed to the attention module.

Inspired from [39], attention masks generated from the attention module can be thought of as spatial probability maps. These spatial probability maps can be learnt using fully convolutional networks. Hence, attention module comprises of a small fully convolutional network of 4 layers.

As shown in Fig. 5(c), the attention module consists of $Conv(4*m, 3), Conv(4*m, 3), Conv(8*m, 3)$ and $Conv(2*m, 3)$. The final output with $2*m$ channels has been split into two units: $Attn^t_{DEM}$ and $Attn^t_{RGB}$ of m channels each which in turn act as an attention mask for the input features F^t_{RU} and F_{RGB}, respectively. Unlike [39], we use multi channel attention maps. We then use element wise channel multiplication to get a weighted set of features with attention channels. A channel-wise summation then fuses the two sets of features together into, F^t_{fused}, the final output of AFM as shown in Eq. (1).

$$F^t_{fused} = F^t_{RU} * Attn^t_{DEM} + \gamma * F_{RGB} * Attn^t_{RGB} \tag{1}$$

where a learnable parameter γ is used for stable learning. γ has been initialized with 0 so as to focus on F_{RU} first and adaptively move the attention to F_{RGB}. To implement an iterative feedback, we store F^t_{fused} over current step and then concatenate it with F_{DEM} to be processed in next step as part of feedback. For the first step, i.e. at $t = 0$, as there will not be any F_{fused}, we use F_{DEM} itself as feedback information for step $t = 0$. We forward F^t_{fused} as input to the reconstruction block. Residing inside the feedback module, we let the attention maps to refine themselves as the iterations proceed. This timely varying attention

units for same input also makes our attention module unique and different from [39].

We run the AFM module for T number of steps. For each step t, we get one set of features F^t_{fused}, which is improved version of itself as the iteration goes on.

We implement **Reconstruction Block** with two units of convolutional layers $Conv(m, 3)$ and $Conv(1, 3)$. For each step of the feedback unit, the reconstruction block takes in F^t_{fused} and produces a residual map denoted by I^t_{res}. The I^t_{res} are the higher frequency details we are interested in generating. We add this residual, I^t_{res} to DEM_{ILR} which we forward from input directly via a global skip connection shown in Fig. 3. The predicted super-resolved DEM at time step t is given by Eq. (2).

$$SR^t = I^t_{res} + DEM_{ILR} \quad \forall t \in \{1, T\} \tag{2}$$

With a recursion of depth T, for each step of t for single data instance, we get one SR, forming an array of predicted SRDEMs with increasing amount of details.

We use $L1$ loss over HRDEM and SR^t for $t \in \{1, T\}$ as given by Eq. (3).

$$L = \sum_{t=1}^{T} |HRDEM - SR^t| \tag{3}$$

The final loss L will be used for back-propagation and training the parameters.

4 Experimental Setup

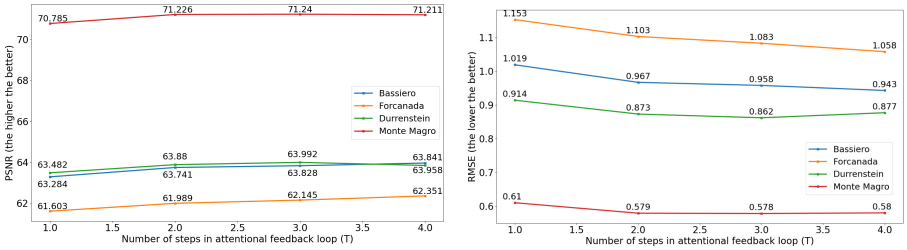
4.1 Datasets

Our goal in this study is to selectively utilize the information from other modalities like aerial images. For fair comparison with existing methods such as [12] and [11], we train our model using dataset provided by Institut Cartogràfic i Geològic de Catalunya (ICC) [40] and Südtiroler Bürgernetz GeoKatalog (SBG) [41]. The terrains provided by these institutes have been pre-processed by the authors of [12]. The dataset used for training comprises of geo-registered pairs of DEM and aerial images of several mountain regions named Pyrenees and Tyrol. DEM patches with a resolution of 2m/pixel have been used as ground truth (HRDEM) elevation maps. These HRDEMs have been downsampled to 15m/pixel to create a corresponding LRDEM. For convenient training, original DEM tiles have been split into patches of size 200x200 pixels, where each pixel intensity signifies terrain height. To effectively avail the aerial information, the resolution of aerial image has been set twice that of DEM, resulting in patches of size 400x400. From all the patches, 22000 patches have been chosen for training and 11000 patches for validation. Also, two regions from Pyrenees namely Bassiero and Forcanada, and two regions from Tyrol namely Durrenstein and Monte Magro have been set aside for testing the network performance. We suggest the reader to refer [12] for more details about the dataset.

4.2 Implementation Details

In this section, we explain the hyper-parameters and details about our experimental setup. We have used convolutional layers with kernel size of 3×3, unless explicitly stated. The parameters in these layers were initialized with *Kaiming* initialization protocol. All the convolutional layers are followed by PReLU activation. For the RGB branch in FE module, we have used first two convolution layers (pre-trained on ImageNet dataset) from VGG-16 network. Later, we allow to fine-tune their weights so as to adapt the weights according to DEM modality. We set m (the number of base channels) to 64 and T(number of steps in feedback loop) to 4. We have chosen T to be 4, as the gain performance in terms of PSNR and RMSE (Shown in Fig. 6) is getting stagnant around $T = 4$. We use N, i.e. the number of residual units, as 16. Since we have used LRDEM with resolution of 15 meters (as stated in [11]), the effective super-resolution factor in our case is 7.5X. We have used a batch size of 4, the max supported with our 4 NVIDIA-1080Ti GPUs. We used learning rate of $\eta = 0.0001$ with multi-step degradation by parameter 0.5 with epoch intervals at [45,60,70]. Parameters were updated with *Adam* optimizer. We have implemented our network in PyTorch framework. After convergence of the network, the value learned by γ is 0.358.

During testing, similar to [11], we have adopted the technique of overlapped prediction with overlap of 25% on all sides of the patch.



(a) PNSR (b) RMSE

Fig. 6. Choice of parameter T (number of steps)

5 Results and Discussions

We use standard root mean squared error(RMSE) and peak signal-to-noise ratio (PSNR) metrics to compare the performance of our proposed method with existing SOTA methods, namely FCN[12] and DSRFB [11]. While RMSE helps understand the cumulative squared error between the prediction and ground truth, PSNR helps to gain the measure of peak error, PSNR and RMSE are complementary measures to compare the performance of SR methods. We also compare the performance with a variant of FCN, FCND which does not use aerial imagery as complementary source of information.

From Table 1, we can infer that our network AFN outperforms both FCN and DSRFB. Using the overlapped prediction, our variant, AFNO performs even

better. Similar observation can be made from Table 2, where AFN has the best PSNR even without using overlapped prediction. Even though the quantitative performance in some areas seems marginal, the gains achieved by our method (over SOTA) in terms of absolute height values are around 0.5 to 1.0m which is quite valuable.

Table 1. Comparison: RMSE values(in meters. The lower the better).

Input	Only LRDEM				LRDEM and RGB		
Region	Bicubic	DSRFB	DSRFO	FCND	FCN	AFN	AFNO
Bassiero	1.406	1.146	1.091	1.083	1.005	**0.943**	**0.926**
Forcanada	1.632	1.326	1.2702	1.259	1.097	**1.058**	**1.030**
Durrenstein	1.445	0.957	0.884	0.868	0.901	**0.877**	**0.854**
Monte Magro	0.917	0.632	0.589	0.581	0.587	**0.580**	**0.566**

Table 2. Comparison: PSNR values (The higher the better).

Input	Only LRDEM				LRDEM and RGB		
Region	Bicubic	DSRFB	DSRFO	FCND	FCN	AFN	AFNO
Bassiero	60.5	62.261	62.687	62.752	63.4	**63.958**	**64.113**
Forcanada	58.6	60.383	60.761	60.837	62.0	**62.351**	**62.574**
Durrenstein	59.5	63.076	63.766	63.924	63.6	**63.841**	**64.061**
Monte Magro	67.2	70.461	71.081	71.196	71.1	**71.211**	**71.417**

From our test regions, we pick one patch each based on certain geographical property, typically containing one major terrain feature. In Fig. 7, first row shows the aerial view of the selected terrain patches. From Bassiero, we select a highly varying terrain patch. From Forcanada, we choose a patch with bare surface. Patches from Durrenstein and Monte Magro respectively have terrains covered with dense vegetation and snow. From comparison results in Fig. 7, we can see that, for Bassiero, our method is able to recover most of the terrain variations in the terrain. In low resolution input of Forcanada, almost all terrain details have been lost, yet our method can recover most of the lost structure. In cases of covered terrains in Durrenstein and Monte Magro, where LRDEM is seen to have more variations, our method has introduced the least noise. Additional results are available in the supplementary video.

5.1 Ablation Studies

To justify the effectiveness of the Attention module, we thoroughly test our network by creating its variants around Attentional Feedback Module. We discuss four major studies in this section.

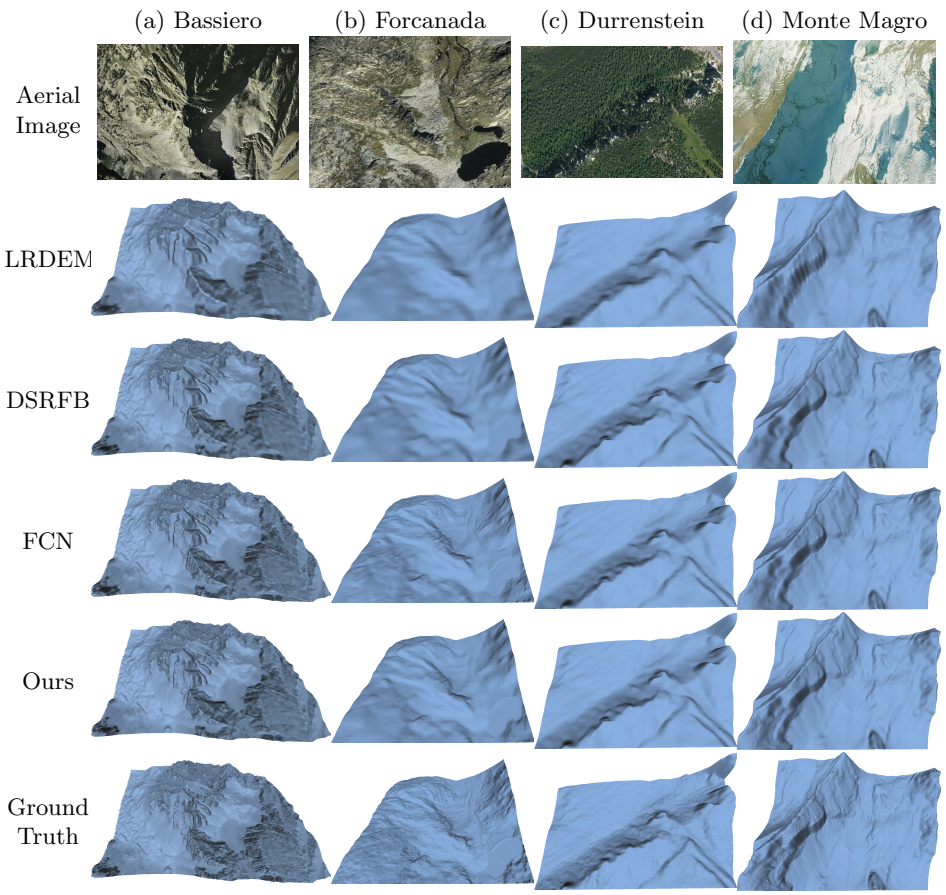


Fig. 7. Qualitative comparison of different DEM super-resolution methods

Without Attention Module: In this experiment, we remove the attention module from the network entirely. For fusing the features from two modalities, i.e. F_{DEM} and F_{RGB}, we use channel concatenation followed by $Conv(m, 1)$ layer. We keep the rest of the setup same as in AFN. The reduction in performance of the network can be seen in Table 3 which supports the role of attention module in selective feature extraction.

Static Attention Masks: In AFN, the attention masks for both F_{RU} and F_{RGB} get updated with iterations. In this study, we move the attention module outside the feedback network and use feedback module only for refining the F_{RU} features. So in this case, we denote the attention state as static and call this variant as AFN0. Comparison from Table 3 confirms that iterative attention can help the network learn more refined feature than fixed attention mask.

Number of channels in Attention Module: To understand the contribution of AFM in performance gain, we changed hyper-parameters. We reduced

Table 3. Ablation Studies.

Region	Without AFM		AFN0		AFN64		AFN	
	PSNR	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR	RMSE
Bassiero	62.406	1.128	63.108	1.04	63.724	0.969	63.958	0.943
Forcanada	60.537	1.303	61.355	1.186	62.141	1.084	62.351	1.058
Durrenstein	62.994	0.967	63.769	0.884	64.116	0.85	63.841	0.877
Monte Magro	70.365	0.64	70.934	0.599	71.154	0.584	71.211	0.58

the number of channels to 64 throughout the attention module. We denote AFN in this setup as AFN64. The proportional reduction in performance reflects the role of AFM in capturing the higher frequency details.

Table 4. Performance of AFND i.e. AFN without using aerial images.

Region	PSNR (in dB, the higher the better)				RMSE (in meters, the lower the better)			
	Bicubic	FCND	DSRFB	AFND	Bicubic	FCND	DSRFB	AFND
Bassiero	60.5	62.261	62.687	62.404	1.406	1.146	1.091	1.128
Forcanada	58.6	60.383	60.761	60.504	1.632	1.326	1.2702	1.308
Durrenstein	59.5	63.076	63.766	63.394	1.445	0.957	0.884	0.923
Monte Magro	67.2	70.461	71.081	70.768	0.917	0.632	0.589	0.611

Performance without Aerial Imagery: To test the flexibility and limitations of AFN, we study its performance in absence of aerial imagery. Getting aligned pair of aerial image and DEM could be challenging sometimes and hence we analyze the performance of AFN in absence of aerial image. In this exercise, we replace the input aerial image with an uniform prior image of same dimensions. We call this variant as AFND. Table 4 shows that despite trained with RGB images, while prediction, AFND selectively picks information from DEM modality and perform consistently better than FCND and almost comparable to DSRFB. Of course, DSRFB was designed to work without RGB. The marginal decrease in performance of AFND compared with DSRFB can be attributed partially to the uniform prior acting as noise and causing the attention module to generate a biased attention response.

6 Conclusion

We have proposed a novel terrain amplification method called AFN for generating the DEM super-resolution. It uses low-resolution DEM and complementary information from corresponding aerial image by computing an attention mask from the attention module along with the feedback network to enhance the performance of the proposed architecture. While this architecture is able to learn well across different terrains, there is a need to further enhance some key features of terrains, especially in regions with high frequency. Hence, there might be a need to explore the use of multi-scale fusion as an extension to the proposed AFN. Also, similar to other computer vision applications, it might be interesting to generate high-resolution DEM using only aerial image as an input.

References

1. Guérin, E., Digne, J., Galin, E., Peytavie, A.: Sparse representation of terrains for procedural modeling. In: Computer Graphics Forum. Volume 35., Wiley Online Library (2016) 177–187
2. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: image and video synthesis using graph cuts. ACM Transactions on Graphics (ToG) **22** (2003) 277–286
3. Musgrave, F.K., Kolb, C.E., Mace, R.S.: The synthesis and rendering of eroded fractal terrains. ACM Siggraph Computer Graphics **23** (1989) 41–50
4. Zhou, H., Sun, J., Turk, G., Rehg, J.M.: Terrain synthesis from digital elevation models. IEEE transactions on visualization and computer graphics **13** (2007) 834–848
5. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: European conference on computer vision, Springer (2016) 391–407
6. Kim, J., Kwon Lee, J., Mu Lee, K.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 1637–1645
7. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2018) 2472–2481
8. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision, Springer (2016) 694–711
9. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Loy, C.C.: Esrgan: Enhanced super-resolution generative adversarial networks. Computer Vision – ECCV 2018 Workshops (2019) 63–79
10. Li, Z., Yang, J., Liu, Z., Yang, X., Jeon, G., Wu, W.: Feedback network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 3867–3876
11. Kubade, A., Sharma, A., Rajan, K.S.: Feedback neural network based super-resolution of dem for generating high fidelity features (2020)
12. Argudo, O., Chica, A., Andujar, C.: Terrain super-resolution through aerial imagery and fully convolutional networks. In: Computer Graphics Forum. Volume 37., Wiley Online Library (2018) 101–110
13. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning. (2015) 2048–2057
14. Galin, E., Guérin, E., Peytavie, A., Cordonnier, G., Cani, M.P., Benes, B., Gain, J.: A review of digital terrain modeling. In: Computer Graphics Forum. Volume 38., Wiley Online Library (2019) 553–577
15. Perlin, K.: An image synthesizer. ACM Siggraph Computer Graphics **19** (1985) 287–296
16. Génevaux, J.D., Galin, É., Guérin, E., Peytavie, A., Benes, B.: Terrain generation using procedural models based on hydrology. ACM Transactions on Graphics (TOG) **32** (2013) 1–13
17. Schneider, J., Boldte, T., Westermann, R.: Real-time editing, synthesis, and rendering of infinite landscapes on gpus. In: Vision, modeling and visualization. Volume 2006. (2006) 145–152

18. Gain, J., Marais, P., Straßer, W.: Terrain sketching. In: Proceedings of the 2009 symposium on Interactive 3D graphics and games. (2009) 31–38
19. Hnaidi, H., Guérin, E., Akkouche, S., Peytavie, A., Galin, E.: Feature based terrain generation using diffusion equation. In: Computer Graphics Forum. Volume 29., Wiley Online Library (2010) 2179–2186
20. Génevaux, J.D., Galin, E., Peytavie, A., Guérin, E., Briquet, C., Grosbellet, F., Benes, B.: Terrain modelling from feature primitives. In: Computer Graphics Forum. Volume 34., Wiley Online Library (2015) 198–210
21. Benes, B., Forsbach, R.: Layered data representation for visual simulation of terrain erosion. In: Proceedings Spring Conference on Computer Graphics, IEEE (2001) 80–86
22. Cordonnier, G., Galin, E., Gain, J., Benes, B., Guérin, E., Peytavie, A., Cani, M.P.: Authoring landscapes by combining ecosystem and terrain erosion simulation. ACM Transactions on Graphics (TOG) **36** (2017) 1–12
23. Guérin, É., Digne, J., Galin, É., Peytavie, A., Wolf, C., Benes, B., Martinez, B.: Interactive example-based terrain authoring with conditional generative adversarial networks. Acm Transactions on Graphics (TOG) **36** (2017) 1–13
24. Allebach, J., Wong, P.W.: Edge-directed interpolation. In: Proceedings of 3rd IEEE International Conference on Image Processing. Volume 3., IEEE (1996) 707–710
25. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. International journal of computer vision **40** (2000) 25–47
26. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. IEEE Computer graphics and Applications **22** (2002) 56–65
27. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 5197–5206
28. Gu, S., Zuo, W., Xie, Q., Meng, D., Feng, X., Zhang, L.: Convolutional sparse coding for image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1823–1831
29. Tai, Y.W., Liu, S., Brown, M.S., Lin, S.: Super resolution using edge prior and single image detail synthesis. In: 2010 IEEE computer society conference on computer vision and pattern recognition, IEEE (2010) 2400–2407
30. Sun, J., Xu, Z., Shum, H.Y.: Image super-resolution using gradient profile prior. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2008) 1–8
31. Kim, K.I., Kwon, Y.: Single-image super-resolution using sparse regression and natural image prior. IEEE transactions on pattern analysis and machine intelligence **32** (2010) 1127–1133
32. Wang, Z., Liu, D., Yang, J., Han, W., Huang, T.: Deep networks for image super-resolution with sparse prior. In: Proceedings of the IEEE international conference on computer vision. (2015) 370–378
33. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. Lecture Notes in Computer Science (2016) 391–407
34. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 4681–4690
35. Zhang, K., Sun, M., Han, T.X., Yuan, X., Guo, L., Liu, T.: Residual networks of residual networks: Multilevel residual networks. IEEE Transactions on Circuits and Systems for Video Technology **28** (2017) 1303–1314

36. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 3147–3155
37. Zamir, A.R., Wu, T.L., Sun, L., Shen, W.B., Shi, B.E., Malik, J., Savarese, S.: Feedback networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 1308–1317
38. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
39. Li, G., Xie, Y., Lin, L., Yu, Y.: Instance-level salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 2386–2395
40. ICC: Institut cartogràfic i geològic de catalunya (icc). http://www.icc.cat/vissir3 (2020) Online Accessed: February 2, 2020.
41. SBG: Südtiroler bürgernetz geokatalog (sbg). http://geokatalog.buergernetz.bz.it/geokatalog (2020) Online Accessed: February 2, 2020.