Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

b-it Bonn-Aachen International Center for Information Technology

Deutsches Zentrum
für Luft- und Raumfahrt
German Aerospace Center

Master's Thesis

Deep Learning based Super Resolution of Urban Digital Surface Models

Krishna Teja Nallanukala

Submitted to Hochschule Bonn-Rhein-Sieg,
Department of Computer Science
in partial fullfilment of the requirements for the degree
of Master of Science in Autonomous Systems

Supervised by

Prof. Dr. Sebastian Houben
Prof. Dr. Paul G. Plöger
Dr. rer. nat. Ksenia Bittner

September 2023

I, the undersigned below, declare that this work has not previously been submitted to this or any other university and that it is, unless otherwise stated, entirely my own work.

Date

Krishna Teja Nallanukala

Abstract

Digital Surface Model (DSM), characterized by their ability to represent both natural and man-made features with precision, plays an indispensable role in diverse fields such as urban planning, environmental monitoring, disaster management, and infrastructure development. However, the intrinsic limitations of traditionally collected DSMs, which capture only a fraction of the Earth's complexity, necessitate the development of super-resolution techniques. This research is dedicated to advancing the field of urban DSM super-resolution through deep learning. The primary objective is the generation of highly accurate high-resolution DSMs, an area of study that remains relatively underexplored in comparison to image super-resolution or DTM super-resolution. The complexity of urban topography, the continuous data, and the presence of high-frequency features in DSMs compared to Digital Terrain Models (DTMs) pose unique challenges. Prior works in super-resolution, typically designed for DTMs, may not fully address the nuances of high-resolution DSM reconstruction.

To bridge this gap, this research investigated the performance of state-of-the-art Generative Adversarial Network (GAN)-based deep learning algorithms such as D-SRGAN, ESRGAN, Real-ESRGAN, Pix2Pix(U-Net), and EfficientNetv2 for super-resolving DSM. These algorithms serve as the foundation for establishing a baseline model for DSM super-resolution. Comprehensive qualitative and quantitative analyses conducted in this research reveal that D-SRGAN stands out as the promising baseline model by performing better than other deep-learning models and classical bicubic upsampling. However, the model couldn't reconstruct the fine details present in the urban environment. Therefore the research focused on the development of a deep learning model with D-SRGAN as a base to improve the baseline model performance, which includes D-SRGAN with multi-head attention layers, channel attention, co-learning architecture, and Encoder-decoder style D-SRGAN. These models do not yield significant improvements over the baseline. This outcome is attributed to the distinctive attributes of DSM, including the absence of high-frequency features in 4x low-resolution DSM and the presence of complex high-level semantics. Moreover, these models demonstrate limited feasibility in enhancing resolution beyond a 4X scale.

Acknowledgements

Foremost, I would like to convey my wholehearted gratitude to my supervisors Prof.Dr.Sebastian Houben, Prof.Dr.Paul G. Plöger, and Dr.rer.nat.Ksenia Bittner for the continuous support of my Research project, and for their patience, enthusiasm, immense knowledge, and motivation. Their guidance helped me all the time with this project. I thank my colleagues at Deutsches Zentrum für Luft- und Raumfahrt: Daniel Panangian, Sandeep Kumar Jangir and Philipp Schuegraf for their continuous support and discussion. I am blessed to have the moral support provided by my parents and friends, which motivated me to push the limits.

Contents

List of Figures

List of Tables

1

Introduction

1.1 Motivation

The Earth's surface is a complex, uneven topography that contains a plethora of knowledge essential to numerous scientific, engineering, and environmental management disciplines [1]. For both researchers and practitioners, it has been difficult to comprehend and effectively represent this terrain. Elevation models are essential in the field of geospatial analysis for understanding and utilizing the complex terrain of our globe. Whether for environmental conservation, disaster management, infrastructure construction, or scientific investigation, the landscape we travel across is anything but flat. Mountains, valleys, rivers, and plateaus are some of the intricate elements that characterize the Earth's terrain in particular ways [1]. Elevation models have become essential tools for understanding these variations and utilizing the potential of geospatial information. An elevation model, also known as a digital elevation model (DEM) is a representation of the Earth's surface, typically in the form of a grid or raster, with each cell containing elevation data [1]. These models have witnessed remarkable advancements over the years, evolving from rudimentary representations to highly accurate and detailed renditions of the Earth's terrain.

Elevation models are becoming more and more crucial to a wide range of applications that benefit society as we progress further into the digital era [2]. As a result of the exponential increase in demand for high-quality elevation data, data-gathering technology, modeling algorithms, and data transmission platforms have all advanced. Furthermore, new opportunities for comprehending the dynamic landscapes of our globe have been opened up by the integration of elevation models with other geospatial information, such as satellite photography and remote sensing data [2]. Basically, these elevation models are of 2 types Digital Terrain Model(DTM) and Digital Surface models (DSM). DTM models represent the elevation of the bare earth terrain, whereas DSMs represent elevation data including all objects on the Earth's surface, including natural features like mountains, valleys, and bodies of water, as well as human-made structures such as buildings, roads, and bridges [2] as shown in Figure 1.1. This thesis mainly focuses on dealing with the DSM data.

DSMs have a wide range of uses that cut beyond conventional lines and transform how we think about the Earth's surface. Some significant fields where DSMs have significantly contributed include Civil Engineering and Infrastructure Development, Natural Resource Management, Disaster Management, Archaeology and Cultural Heritage Preservation, Geological Studies, Ecological Research, and Urban

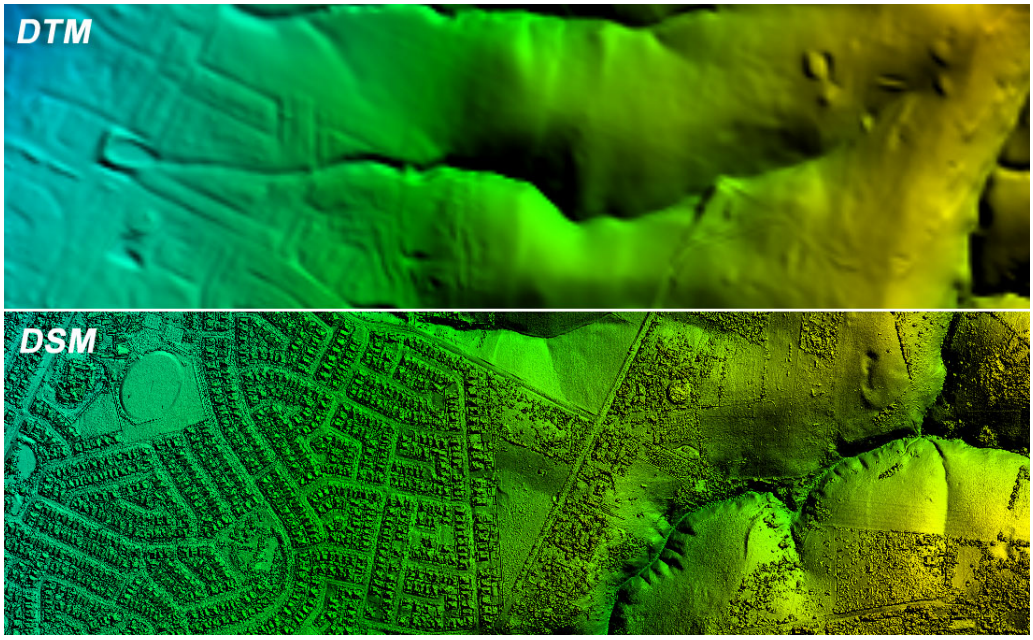


Figure 1.1: Illustration of difference between DTM and DSM [3]

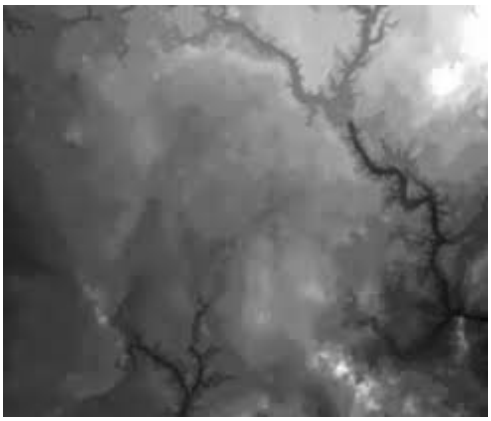
Growth and Land Use Planning [4–6]. Due to the broad level of applications requiring these DSM models, DSMs' accuracy and resolution are crucial elements, and demand for such high-resolution DSM has increased. DSMs are produced using a variety of methods, but mostly through the use of remote sensing tools like LiDAR (Light Detection and Ranging) and photogrammetry. For instance, LiDAR measures the amount of time it takes for a laser to return after striking the ground. It generates laser pulses from an aerial or terrestrial platform. Combining and processing these exact measurements results in a dense point cloud, from which the DSM is produced. Contrarily, photogrammetry uses overlapping photos taken at various angles to triangulate the locations of points on the surface of the Earth. Once more, these points are combined to produce the DSM, which yields a remarkably accurate depiction of the topography. However, the limits of data acquisition technology, the financial cost of acquisition, and the computing capabilities available at the time of their development have frequently limited the geographic resolution of these DSMs [6] [4] [1]. As a result, it has become harder to detect tiny surface characteristics or distinguish fine-scale details. Generally, the resolution of elevation models is measured using the ground sampling distance (GSD). The lower the GSD value higher the resolution of the data. For example, a 30m GSD means the distance between each pixel covers 30m of the earth's surface.

This research mainly focuses on urban areas, where the availability of high-resolution data is very limited and challenging to collect due to the complex nature of the urban environment [7]. Whereas, the low-resolution DSMs lack fine details and portrayal of the complex urban surface topography, limiting their use for many applications. Within this context, the field of Super-Resolution, a technique aimed at enhancing the spatial resolution and fidelity of data, represents a significant breakthrough in geospatial technology.

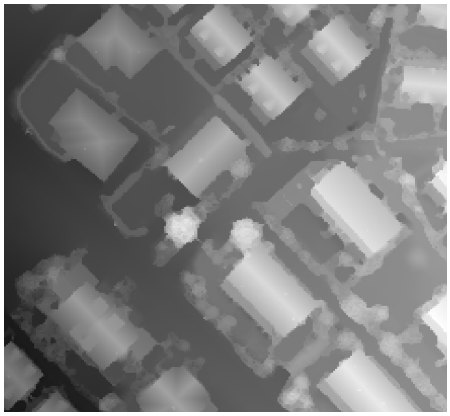
Super-resolution (SR) is a well-researched topic in computer vision, where it achieves reconstructing high-resolution images by employing innovative techniques that fuse information from multiple data sources or apply sophisticated algorithms for a single source to enhance the clarity and precision of the image [6]. DSM grid could be considered as an image because of their structural similarities where their planar coordinates and elevation values can be seen as corresponding pixel position and intensity values of an image [8] but the range of pixel intensities is different from the range of elevation values. Images are 8-bit files but elevation models are 32-bit raster files [8]. Therefore, the methodologies employed in DSM Super-Resolution are diverse, drawing inspiration from fields such as image super-resolution. Additionally, the classical upsampling techniques fail to enhance the clarity of high resolution/low GSD DEM [4]. Therefore, this thesis aims to investigate and develop deep learning-based advanced SR techniques inspired by image SR and DTM SR for enhancing the spatial resolution of DSMs in urban areas, with a focus on improving the accuracy and precision of height information.

1.2 Problem Statement

Super-resolution (SR) techniques have become increasingly important as the demand for high-resolution data has grown. The SR algorithms are categorized into 3 types: Interpolation-based, reconstruction-based, and learning-based [4]. The Interpolation techniques improve the resolution by estimating the missing elevation values based on information available in the neighboring evaluation values. The resulting high-resolution data lack high-frequency topographic details and lead to blurring. Reconstrucion-based algorithms require extra information such as elevation points, contours, etc., to restore the lost features. Even though they produce better results than interpolation-based techniques, these fail to work for large magnification factors and difficult to obtain additional information in large-scale applications [4].



(a) Digital Terrain Model (DTM)



(b) Digital Surface Model (DSM)

Figure 1.2: Visualization of both DTM and DEM

Lastly, learning-based models construct a mapping function between low-resolution and high-resolution data to learn and produce high-resolution information. Since the development of deep learning the application of deep learning for SR has become a hot topic among researchers. However, most of the

research in deep learning-based SR of elevation models is applied to digital terrain models(DTM) or natural topography, and most of the architectures are based on image SR techniques. The research on DSM SR has not yet been explored much and the existing DTM/image SR deep learning techniques might not work efficiently due to challenges introduced by the complex urban topography, continuous data, and high-frequency features [7]. Figure 1.2 shows the difference between DTM and DSM features. Thus, this calls for the need for such sophisticated models for reconstructing high-resolution urban DSMs. In this thesis project, we aim to address the SR of DSM based on deep learning, and the proposed method will be evaluated using both qualitative and quantitative measures, and their effectiveness will be compared with others to demonstrate their performance in improving the resolution of urban DSMs. This research will contribute to the development of accurate and reliable high-resolution DSMs for urban areas that can benefit a wide range of applications.

1.3 Report Outline

This report is composed of six chapters; Chapter 1 provides the motivation for this research work, followed by a problem statement and a report outline. Chapter 2 enlightens the necessary knowledge about the concepts used in this research work. Chapter 3 provides SOTA work for image super-resolution and elevation models super-resolution. Chapter 4 presents the methodology, datasets, and evaluation metrics. Chapter 5 presents the experiments conducted and their results. Chapter 6 discusses the conclusion, contributions, lessons learned, and future research direction.

2

Background

This chapter describes some basic concepts that are involved in this research to understand the later chapters.

2.1 Deep learning based super-resolution

Image super-resolution aims to extract the equivalent HR images from the LR images. The LR image I_x is typically modeled as the result of the degradation described below [9]:

$$I_x = D(I_y; \delta) \tag{2.1}$$

where D stands for a degradation mapping function, I_y is the associated HR image, and δ stands for the degradation process's parameters (such as noise or scaling factor). Typically, only LR images are provided and the degradation process (i.e., D and δ) is unclear. Researchers must extract an HR approximation of the ground truth HR picture $\hat{I_y}$ from the LR image in this scenario, by doing the following [9]:

$$\hat{I_y} = F(I_x; \theta) \tag{2.2}$$

where F is the super-resolution model and θ denotes the parameters of F.Researchers are attempting to predict the degradation mapping despite the fact that the degradation process is unknown and can be impacted by a number of variables (such as compression artifacts, anisotropic degradations, sensor noise, and speckle noise). Most works directly model the degradation as a single downsampling procedure, as follows [9]:

$$D(I_y; \delta) = (I_y) \downarrow, s \subset \delta, \tag{2.3}$$

"where $\downarrow$s is a downsampling operation with the scaling factor s. As a matter of fact, most datasets for generic SR are built based on this pattern, and the most commonly used downsampling operation is bicubic interpolation with antialiasing" [9]. This scenario of estimating the downscaling degradation is known as non-blind super-resolution. This research was based on non-blind single image super-resolution (SISR).

2.2 Bicubic upsampling

One of the simplest and widely used techniques for image upscaling (increasing resolution) is bicubic interpolation. Bicubic interpolation estimates the pixel values in a high-resolution image based on the values in a low-resolution image [4]. The formula for bicubic interpolation is as follows:

Given a low-resolution image represented as a 2D grid with pixel values $f(x, y)$ where x and y are the coordinates in the low-resolution image, and x_i, y_j are the coordinates in the high-resolution image, the bicubic interpolation formula for estimating a pixel value $I(x_i, y_j)$ in the high-resolution image is [4]

$$I(x_i, y_j) = \sum_{m=-1}^{2} \sum_{n=-1}^{2} a_{mn} f(x + m, y + n) \tag{2.4}$$

Here, a_{mn} are the coefficients of the bicubic kernel, which determine the weighting of neighboring pixels in the low-resolution image. These coefficients are typically precomputed and remain constant during the interpolation process. Bicubic interpolation aims to provide a smooth and visually pleasing upscaling of images but is a simple method compared to more advanced super-resolution techniques [4].

3

Related Work

In this chapter, the literature review on various super-resolution techniques used for image and elevation model super-resolution will be presented.

3.1 Image super resolution

The goal of image super-resolution is to recover fine-grained information that might have been lost during image degradation caused by factors such as sensor limitations, compression, or imaging hardware constraints [10]. By leveraging sophisticated algorithms and deep learning techniques, researchers have made remarkable progress in addressing this challenging problem. Over the years, various techniques have been developed for image super-resolution, most of the techniques fall under mainly two categories, single Image super-resolution, and Multi-image super-resolution [11]. The single image super-resolution (SISR) technique only uses a single low-resolution image to generate the high-resolution counterpart. Whereas, Multi-image super-resolution (MISR) leverages information from multiple low-resolution images to generate a super-resolved image. Based on the application and availability of resources, the researchers utilize either of two approaches. In this research work the application is to use only single LR information to generate a HR counterpart. Therefore, most of the related work discussion is based on SISR techniques. Apart from these two main categories based on resources, the SISR techniques fall under two categories based on the underlying principles. Traditional and deep learning techniques are those two principal categories. Traditional approaches often employ techniques such as interpolation, spatial-domain filtering, or statistical-based methods to upscale the images [4]. Nearest Neighbour, Bilinear, and Bicubic are the interpolation algorithms. These interpolation methods, while simple and computationally efficient, tend to produce blurry results lacking fine details [4]. Figure 3.1 gives a brief overview of related work for single image super-resolution.

Additionally, filtering-based methods, such as Wiener filtering and total variation regularization, aimed to recover high-frequency information by exploiting image statistics or minimizing certain energy functions. While these techniques showed promise in enhancing image quality, they struggled with preserving complex textures and sharp edges [11]. Authors in [4] compared the interpolation techniques with learning-based techniques developed for image SR for SR of DTM models. Their experimental results proved that neural network SR-based methods using a learning strategy improved the resolution. In recent years, due to the emergence of deep learning techniques, various architectures have been used for computer vision

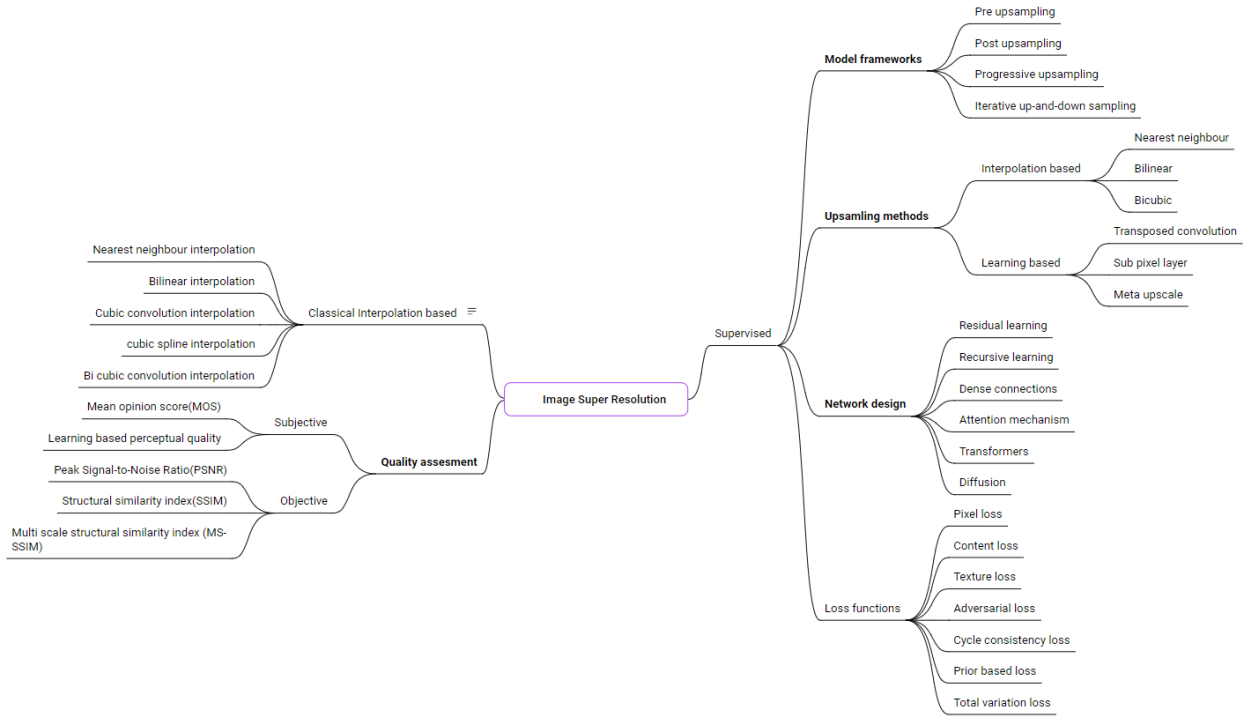


Figure 3.1: Brief overview of related work for Image super resolution

tasks, particularly Convolutional Neural Networks(CNN) which have revolutionized the field of computer vision. CNN-based models showed exceptional results in generating high-resolution images by learning non-linear mappings between low-resolution and high-resolution pairs [11]. These deep learning models can recognize complex correlations and patterns in the data, resulting in more accurate and aesthetically pleasing outcomes.

Authors in [12] introduced the first deep learning-based SR model called SRCNN, which resulted in better results when compared with interpolation methods. The SRCNN model contains 3 convolutional layers and 2 ReLu layers stacked together. The LR input to this SRCNN network is first downsampled and later upsampled using the Bicubic upsampling method as the network architecture is designed to produce the same resolution output as input. Therefore, these designs of networks are known as early upsampling networks [11]. However, as the size of the network is small it fails to learn more features and reconstruct the details present in HR image [6]. Later, the network size is increased by adding one extra convolutional layer and a deconvolutional layer to the design of the SRCNN network by authors of FSCRNN [13]. Another aim of this FSRCNN network is to improve the computational speed to 24fps compared to 1.3fps of SRCNN. These modifications to the network improved the quality and speed [13]. Unlike SRCNN this is a late upsampling design network, where the input to the network is the original LR patch without upsampling [13]. Subsequent architectures built upon this SRCNN foundation and developed more sophisticated models. Authors in [14] designed a very deep network called Very Deep Super

Resolution(VDSR) unlike SRCNN and FSRCNN to extract better features and to learn generalizable representation for the reconstruction of HR efficiently.VDSR is an early upsampling network similar to the SRCNN network. Efficient sub-pixel convolutional neural network(ESPCN) [15] uses the subpixel convolution in the very end to upsample and perform feature extraction on low-resolution data, This process significantly reduces the requirement for memory and computation power [11]. All these above mentioned networks are linear networks without any skip connections, just the convolutions are stacked on top of each other [11]. Several other networks belong under this category of linear networks such as DnCNN [16], and Image Restoration CNN(IrCNN) [17].

Although these deep networks perform SR tasks they are affected by the vanishing gradient problem, which restricts designing very deep networks for better feature extraction [11]. Residual networks are architectural innovation that introduces skip connections, which enable the training of deep neural networks without being affected by vanishing gradient problems. Thus updating the weights of previous layers efficiently. Enhanced Deep Super Resolution(EDSR) [18] is such model proposed for SR based on ResNet architecture, authors demonstrated improvements in performance by removing batch normalization layers of modified ResNet architecture for SR task [11]. Furthermore, the model is extended to work for multiple scales and named Multi-scale Deep SR(MDSR). These networks achieve better performance than older architectures like SRCNN, VDSR, and other architectures closely related to ResNet architecture [11]. Residual networks are of two types single-stage and multi-stage nets [11], the EDSR model is a single-stage network that only uses a single model for the SR task. Whereas, multi-stage nets are composed of several subnets that are typically trained one after the other. FormResNet introduced by authors in [19] is a multistage residual network based on the DnCNN network, both stages are similar to the DcNN network. This network only improved the results by a small margin when compared to the actual DRCNN network [11]. Additionally, all these residual models improved the SR results significantly and future models for this SR task incorporated these residual blocks in network design.

Furthermore, in the search for a perfect architecture for super-resolving the images, researchers utilized recursive networks that incorporate recursively connected CNN layers or linking units recursively. This design's primary goal is to gradually divide the challenging SR problem into a collection of manageable, easier problems. Deep Recursive Convolutional Network(DRCN) [20] is the first recursive network for SR tasks by using the same CNN layers multiple times. Similarly, Deep Recursive Residual Network (DRRN) [21] is a combination of residual and recursive networks, making it deeper than previous models such as VDSR, DRCN, and SRCNN with 52 convolutional layers [11]. Furthermore, a Memory network(MemNet) was designed by authors in [22] to use information from memory similar to how recurrent networks work. So far all the models are typically predicted in one step using CNN, but this may not always be possible for huge scaling factors like 8x or greater. To solve this problem progressive reconstruction networks are designed to super-resolve images in multiple steps, for example, first producing 2x output and later generating 4x [11]. Deep Laplacian pyramid super-resolution network (LapSRN) [23] is one such network that performs 8x upsampling progressively by using three subnetworks sequentially producing 2x, 4x, and finally 8x. The scale residuals produced by each subnetwork are added with the corresponding upsampled bicubic images to generate the final super-resolved images. Another distinction

of this network training with others mentioned above is it uses a differentiable variant of L1 loss known as charbonnier loss which is efficient in handling outliers [23], whereas other methods use the MSE loss(L2) function.

In order to boost the model's performance further, super-resolution algorithms based on densely connected CNN layers were proposed, taking inspiration from DenseNet [24] architecture. The major goal behind this design is to incorporate hierarchical cues that are available along the network depth to obtain great flexibility and richer feature representations. DenseNet architecture uses "dense connections between the layers i.e. a layer directly operates on the output from all previous layers. Such an information flow from low to high-level feature layers avoids the vanishing gradient problem, enables learning compact models, and speeds up the training process" [24]. Residual Dense Network (RDN) [25] combines the residual skip connections from ResNet and dense connections from DenseNet to learn the local patterns from hierarchical feature representations [25]. In this network, residual dense connections are introduced at local and global levels and enable the model to extract better features [11]. However, all the networks discussed above considered feature channels and spatial locations to have equal importance, but in the super-resolution case, not all features have the same importance. "In several cases, it helps to selectively attend to only a few features at a given layer" [11]. In recent years, the use of attention mechanisms coupled with CNN brought significant improvements in the field of computer vision. Residual Channel Attention Network(RCAN) [26] was developed to address the challenge of enhancing the resolution of images while preserving fine details and textures, this network has made significant contributions to the field of single image super-resolution (SISR) [9]. RCAN represents an evolution of the traditional residual network (ResNet) architecture by incorporating channel attention mechanisms to focus on high-frequency information [27]. This network showed better performance than models like VDSR and LapSRN models [11]. However, this model has high computational complexity compared to models like VDSR, MemNet, or LapSRN.

Although these above-mentioned methods restore the resolution they lack the ability to generate realistic or better perceptual quality images and often they produce smoothened images. Generally, these CNN-based models described above try to increase the Peak signal-to-noise ratio (PSNR), which is observed that a higher PSNR value lacks some realism in detail and is too smooth [27]. To restore images with sharp and realistic images, Generative models are designed for image SR tasks. Super-resolution Generative Adversarial Network (SRGAN) [28] was the first GAN-based model for SR reconstruction to recover 4x downsampled images. Generally, a GAN model consists of two parts: a generator and a discriminator. The generator is responsible for generating the data in this SR case generating a super-resolved image and the discriminator is responsible for discriminating whether the generated image is real or fake [28]. These two models are involved in a mini-max game, where one tries to beat the other. The generator tries to fool the discriminator while the discriminator tries to distinguish the difference between real and fake. This learning is known as Adversarial learning and is used as a loss function for the training. The SRGAN uses sub-pixel convolutional layers instead of transposed convolutional layers for upsampling the image after feature extraction. Additionally, the SRGAN network also contains L1 loss and perceptual loss along with adversarial loss. Which makes these networks efficient at generating data. The perceptual

loss is calculated by a pre-trained VGG network.

Another GAN-based model was developed by authors in [29], which is an improved version of SRGAN because of its unpleasant artifacts. SRGAN was improved in 3 aspects: network structure, perceived loss, and adversarial loss. ESRGAN model removed all batch normalization layers and the original structure was replaced with residual-in-residual dense blocks (RRDB). RRDB blocks are inspired by the dense block RDB model [25]. "Removing the BN layer can help improve generalization and performance, and reduce computational complexity" [29]. In addition, the ESRGAN model discriminator is improved by using the relativistic GAN model which estimates the probability that the generated image is more realistic than a fake image. ESRGAN produced results better than the SRGAN model without any artifacts and more realistic images [9]. However, both these generative models tend to "produce less meaningful high-frequency noise that has nothing to do with the input image" [11]. Therefore, the SRFeat [30] model was developed by authors to add a discriminator based on the feature domain. Furthermore, the degradation achieved by bicubic downsampled images would be different from the degradation observed in real LR images. Hence models trained on HR-LR pairs would not perform better in super-resolving real LR images. The model designed to super-resolve such real-world LR images is RealESRGAN [10], which comes under the category of blind super-resolution.

Traditionally, deep learning-based super-resolution methods relied on convolutional neural networks (CNNs) to upscale low-resolution images. While effective, these techniques struggled to capture intricate textures and fine-grained details, leaving room for innovation. Additionally, they focus more on local features and ignore global features [31]. This is where Transformers, with their attention-based mechanisms and ability to capture global context, stepped in to redefine the landscape [31]. Transformers basically use scale dot product type of self-attention, which calculates query, key, and value vectors to establish a relationship between one pixel with all other pixels. This self-attention can model long-term dependencies in images which aids in recovering the textural characteristics of images [31]. Authors in [32] proposed a method to fuse the texture information from reference images with transformer-generated images to fuse different levels of features. [33] presented a hybrid attention transformer that enhances the capacity to investigate pixel information by including channel attention into the transformer and suggesting an overlapping cross attention module (OCAB) to more effectively fuse features from several windows. The efficient transformer (ET) and efficient multiheaded attention (EMHA) techniques were proposed by authors in [34] to capture the long-term dependencies between similar patches in an image to conserve computational resources while enhancing model performance. For super-resolution reconstruction, [35] utilized Swin transformer to create SwinIR which can be utilized to learn the long-term dependencies of images utilizing a shifted window approach.

Another particular network architecture being used in computer vision is The Diffusion model which has gained popularity in recent years due to its effectiveness in producing high-quality images in various applications [36]. The Diffusion Model is a probabilistic generative model used in computer vision for image denoising, image inpainting, and other tasks involving image manipulation and restoration. It is designed to model the process of spreading information (pixels) throughout an image while iteratively adding noise. The core idea behind the Diffusion Model is to simulate the evolution of an image over

multiple steps, where each step corresponds to a discrete-time point [36]. During each step, noise is added to the image, and the goal is to conditionally generate the next image state given the previous state and the added noise. This process is repeated iteratively, gradually transforming a noisy or corrupted image into a clean and denoised version. [36] introduced the Implicit Diffusion Model (IDM) for high-fidelity image super-resolution, The implicit neural representation is used in the decoding process to learn continuous-resolution representation, and IDM blends an implicit neural representation with a denoising diffusion model in a unified end-to-end framework. Another model was introduced for diffusion model is known as ResShift "that proposes a novel and efficient diffusion model for SR that significantly reduces the number of diffusion steps, thereby eliminating the need for post-acceleration during inference and its associated performance deterioration" [37].

3.2 Elevation models super resolution

Elevation Model Super-Resolution" is an emerging field within geospatial science and computer vision that addresses these challenges. It focuses on enhancing the spatial resolution and accuracy of elevation models to provide more detailed and precise representations of the Earth's surface. As discussed in the introduction Chapter **??**, elevation models are of two types DTM and DSM. As discussed in the above section detailing various state-of-the-art methods for image super-resolution ranging from traditional to deep learning methods. Most of the super-resolution algorithms for the elevation models are inspired or derivatives of image super-resolution techniques. Therefore, this section mainly discusses the super-resolution algorithms used for DSM and DTM.

Authors in [8] introduced the first deep learning-based SR model called D-SRCNN for DTM SR which resulted in better results when compared with interpolation methods. However, the D-SRCNN model is designed based on an image-based SRCNN network developed in [12]. Similarly, [38] proposed a deep residual network known as DRNN for DTM super-resolution which was an implementation of DRNN [21] model for image super-resolution. In [39] a deep gradient network (DPGN) was proposed for the DTM task but the model was initially trained on an image super-resolution task and by using transfer learning finetuned for DTM. The idea of this transfer learning is to improve the model reconstruction accuracy by learning the patterns/ features from images. Authors in [40] developed for DTM super-resolution proposed a new model with three subnetworks that a) extracts feature maps, b) infers the high-frequency details, and c) refines the result by combining the low-resolution input with the details from b). This is an alternative to configuring a single network that directly maps low-resolution depth values to high-resolution depth values. Similarly, [1] proposed network improves super-resolution reconstruction outcomes by using both the internal prior of the particular DEM and the external prior of the DEM dataset. Authors in [41] proposed an approach called the improved double-filter deep residual neural network (EDEM-SR) that combines filters with various receptive field sizes to fuse and extract information and rebuild a higher-resolution DEM that is more realistic. However, as all these models use L1/L2 loss for computing the error between high resolution and generated output, the output of these models will result in a smoothed DTM.

Similar to how the image SR task started using GAN models for generating realistic images, the DTM

SR task also adapted to using GAN training. The first GAN model used for DTM SR is D-SRGAN [6], which is an image SR technique SRGAN [28]. Similarly, DTM Enhanced SRGAN (D-ESRGAN) [42], Conditional generative adversarial network (CEDGAN) [43]. Authors in [4] compared the classical interpolation techniques with SRGAN, ESRGAN, and CEDGAN models developed for image SR for SR of DTM models and their experimental results proved that neural network SR-based methods using learning strategy improved the resolution better. Moreover, the SRGAN model performance in supper resolving is better than the ESRGAN and CEDGAN [4]. Authors in [**?**] proposed a model using Efficientnetv2 for SR of DTM, and their experimental results proved that their model is more efficient than D-SRGAN and classical interpolation methods. Inspired by how the image SR improved the performance by adding an attention mechanism to networks, a D-SRCAGAN [44] method was implemented for DTM SR based on the channel attention SRGAN model for image SR. Transformers or diffusion network methods are not experimented with for DTM super-resolution tasks.

Most of the models discussed in the elevation model SR section focus on DTM SR rather than DSM SR. The main reason is that there is very limited research that has been done on DSM super-resolution. Authors in [2] generated high-resolution DSM by first super-resolving images using image-based SR techniques and later DSM was generated by using multiple images. Here multiple super-resolved images are used for generating a sub-pixel DSM by performing dense image matching. This process is not a typical DSM SR but can be considered indirect super-resolution. [7] proposed the use of a multi-scale network for reconstructing high-resolution urban DSMs and compared it with traditional techniques and proved that their network is effective in improving the resolution of complex urban topography. They used scales of 2x,4x, and 8x for downsampling the high-resolution DSM using the nearest neighborhood and used generated LR and HR samples for the training of the model. The multiscale network consists of ”multiple subnetworks Each of these subnetworks performs a 2-time reconstruction to its input urban DSM” [7]. This is the only direct research on DSM super-resolution using deep learning as far as my literature search. Therefore, this research mainly focuses on investigating the performance of state-of-the-art models for DTM and images on DSM.

4

Methodology

4.1 overview

DSMs have become a fundamental data source for many scientific and engineering applications, providing critical information about the Earth's surface particularly urban areas for a wide range of remote sensing applications. This research work aims to use the deep learning-based models for DSM SR. As discussed in the related work Section 3, many SOTA deep learning architectures were designed for single image SR, ranging from using basic Convolutional Neural Networks (CNN), Generative Adversarial Networks (GANs), Attention-based GANs, Variational autoencoders, Transformer models and Diffusion models. Furthermore from the related work, the existing DTM super-resolution models are developed based on single image SR(SISR) methods, and very few architectures were experimented with for elevation model super-resolution. Apart from that due to the complexity of the urban topography, the continuous nature of data, and the more high-frequency features present in DSM than DTM, the SISR developed for DTM may or may not reconstruct better high-resolution DSM. It is also known that there is only one research paper that worked on the DSM super-resolution as far as my knowledge and that didn't compare their model with other deep learning models. Thus it leaves us with the question of how the SISR methods work in the case of continuous and high-frequency featured data like DSM. Therefore, this thesis aims to investigate the SOTA architectures and their efficiency in super-resolving DSM. Moreover, developing a deep neural network architecture for DSM SR and performing qualitative and quantitative comparisons with various other models. Furthermore, from the various network architectures discussed in the related work of elevation models and image super-resolution, it is evident that GAN-based models are effective in generating sharp and realistic super-resolved outputs. Therefore adversarial way of training is used as a learning principle for this research work. Additionally, according to the project proposal this research aimed to improve the resolution of DSM from 5m GSD to 0.5m GSD. Therefore, this thesis also aims to find the feasibility of achieving such a resolution using single-source super-resolution techniques. The codes and trained models for this research can be found on GitHub repo [45].

4.2 Generative Adverserial Networks

GANs are a type of deep learning model developed to learn and mimic data distributions, allowing the generation of new data instances with properties similar to those of a given dataset [9]. The fact that this

approach can generate incredibly realistic and varied results across a variety of domains, including image synthesis, style transfer, and data augmentation, has attracted a lot of attention. The fundamental idea behind GANs is the competitive training of the generator and discriminator neural networks [9]. The discriminator seeks to distinguish between real and artificially generated data, whereas the generator seeks to produce artificial data that is indistinguishable from actual data. The generation of increasingly convincing data instances results from the generator learning to enhance its output through an adversarial training process. In parallel, the discriminator improves at distinguishing between real and fake data, leading to a dynamic equilibrium where the generator produces extremely realistic samples. In the pursuit of enhancing elevation model quality and detail through super-resolution, Generative Adversarial Networks (GANs) have become a well-known and cutting-edge technique. To overcome the inherent drawbacks of conventional interpolation methods, GANs provide an appealing framework for producing high-resolution images with an impressive level of realism and fidelity. To guide the training of the GAN for super-resolution, we design an ensemble of loss functions that incorporate both content loss and adversarial training principles. Content loss emphasizes the generator to produce the content of the image close to the ground truth. Adversarial loss functions encourage the generator to produce images that are indistinguishable from authentic high-resolution samples, thereby driving the improvement of image quality over training iterations.

The technical flow chart of the GAN model for the super-resolution task is illustrated in Figure 4.1. Here the generator G is a fully convolutional neural network that takes the low-resolution input and generates a super-resolved output. Every GAN network used for super-resolution consists of a few essential components such as feature extractor layers, upsampling layers, and convolutional layers at the end. The data is passed through initial convolutional layers followed by residual blocks which act as feature extraction, and later upsampling layers to increase the spatial resolution of the image. The skip connections in the residual blocks enable the model to learn the difference between low-resolution and high-resolution data efficiently. Further details regarding the generator network will be detailed in experiments Chapter 5. On the other hand, the discriminator consists of convolutional layers and fully connected layers, with a final output layer that produces a single scalar value representing the probability that the input image is real or fake. Generally, a pixel-discriminator evaluates the realism of the entire image as a whole, but it fails to provide fine-grained feedback and doesn't model high frequencies which leads to blurry results [46]. Therefore, this research work uses a Patch discriminator designed by authors in [46] to model high frequencies. This patch discriminator tries to classify if each N $\times$N patch in an image is real or fake as shown in Figure 4.2. This discriminator provides fine-grained feedback on local details, textures, and structures within the image and is computationally more efficient than pixel-based discriminators [46].

As we know from related work most of the models used for DTM SR are modified models of image-based SR. Authors in [4] compared three deep learning models SRGAN, ESRGAN and CEDGAN for DTM SR and their experimental results indicate that the SRGAN network is better for reconstructing high-resolution DTM. Even though ESRGAN is an improved version of SRGAN, it failed to work in the case of DTM. Furthermore, authors in [**?**] compared the EfficientNetv2 model with D-SRGAN [6] for DTM

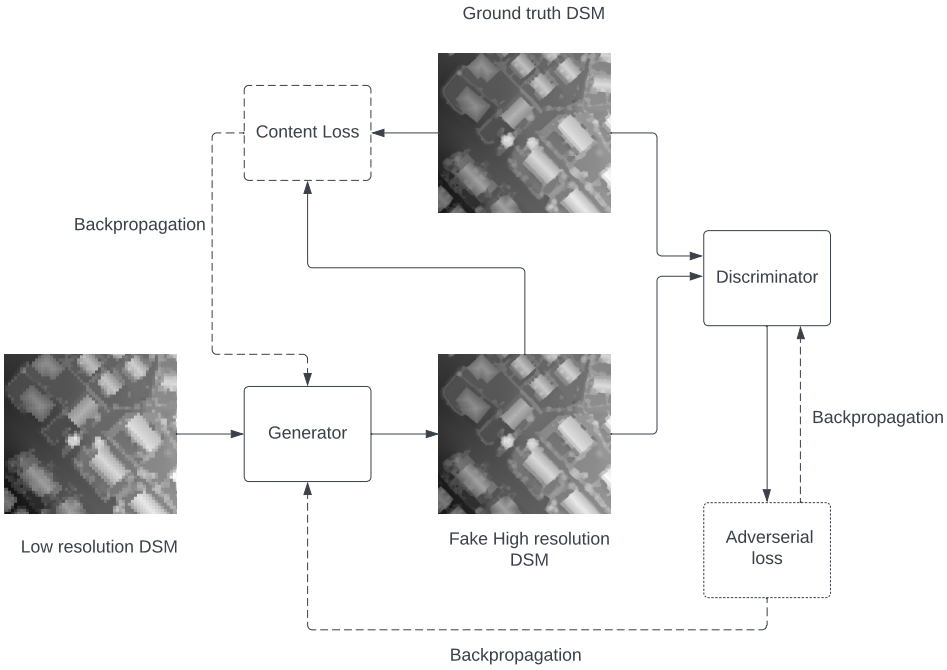


Figure 4.1: The technical workflow of GAN, low-resolution DSM is given as input to the generator and generated fake DSM along with real HR DSM is used by discriminator to calculate the adversarial loss and backpropagated to adjusted the weights to enable the generator to produce SR DSM. Generated image and ground truth are also passed to calculate content loss and backpropagated only to the generator

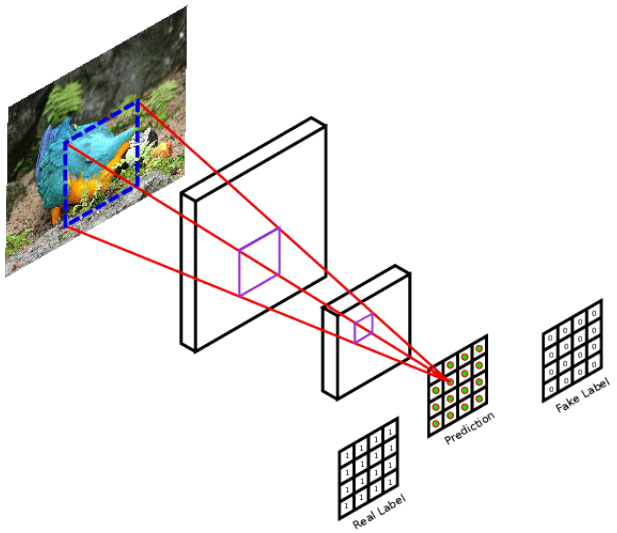


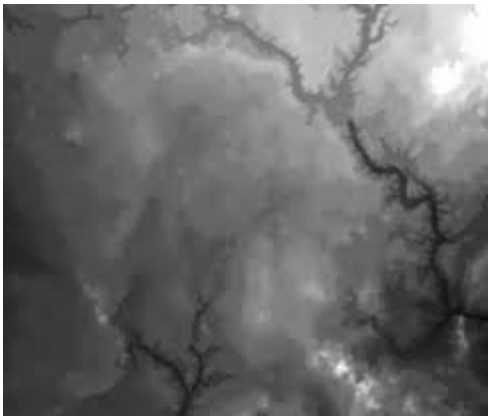
Figure 4.2: PatchGAN discriminator [47]

super-resolution. Meanwhile, the Pix2PixHD [46] model has been the state-of-the-art image-to-image translation model that uses U-net architecture for converting semantic images to photo-realistic images, which can be used in our case for translating a low-resolution input to high-resolution output. Considering these models which worked for DTM SR and no prior work on DSM SR, it is the primary task to establish which architecture works best in the case of DSM. Therefore, initial experiments in this research will be performed to select the best model that worked among D-SRGAN, EfficientNetv2, ESRGAN, and U-net for DSM. Later, the best model will be selected based on both the qualitative and quantitative results to establish a baseline deep learning model and further modifications to the baseline model will be proposed to improve the results.

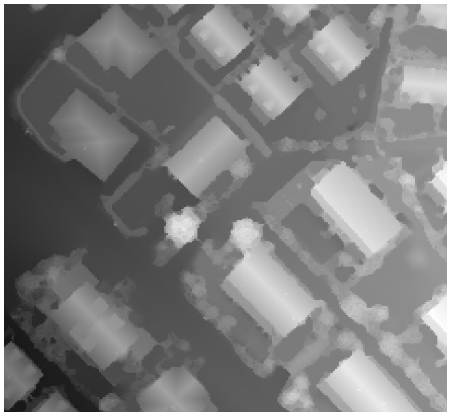
4.3 Dataset description

To investigate the effectiveness of models in improving the resolution of elevation models (DTM & DSM) in this research work, we have used the open-source Switzerland dataset hosted by the Swiss Federal Office of Topography (swisstopo). The HR elevation models of the whole area of Switzerland were collected using airborne LIDAR with a resolution of 0.5m GSD for DSM, and DTM models of 0.5 and 2m GSD. In this thesis work, we have only used data from Zurich canton(state) as shown in Figure 4.4. The elevation files are provided in tif file format; each tile has a dimension of 2000*2000 for 0.5m GSD. For our experiments, each 2000 tile tif file is cropped into equal numbers of 256*256 tiles due to memory issues during training. The dataset contains DSMs in a semi-urbanized area with mixed topographic features including buildings and vegetation. The data of these elevation models is different from normal images because they are 8-bit files whereas, elevation models are 32-bit float data that store the height of a particular data point with respective to sea level. For example, if a pixel in DSM has a value of 430.56 it means that the pixel is 430.56 meters from sea level.

As supervised super-resolution models require an LR-HR pair for training, LR DSM/DTM is generated by downsampling the HR(256*256) using bicubic to scale 64*64 (4x), 128*128(2x). The training dataset uses 1500 tiles from cropped Zurich canton and the test set uses handpicked 200 tiles from the area in Zurich city to test the accuracy of the model specifically in urban centers. The data is normalized using standard deviation because elevation maps of different data sources can have varied elevation ranges. Therefore, they need to be normalized to make the model training easier and stable. Simple Min-Max normalization is sensitive to elevation value outliers [48]. For example, a large value at a single pixel will affect the rest of the elevation map after the Min-Max normalization. Therefore, the data is normalized by calculating the global standard deviation of the trainset and normalizing the dataset using that standard deviation value. Sample images of DTM, DSM, and RGB images being used for this research are illustrated below in Figure 4.3.



(a) Digital Terrain Model (DTM)



(b) Digital Surface Model (DSM)

(c) RGB image

Figure 4.3: Visualization of DTM, DSM, and orthoimage

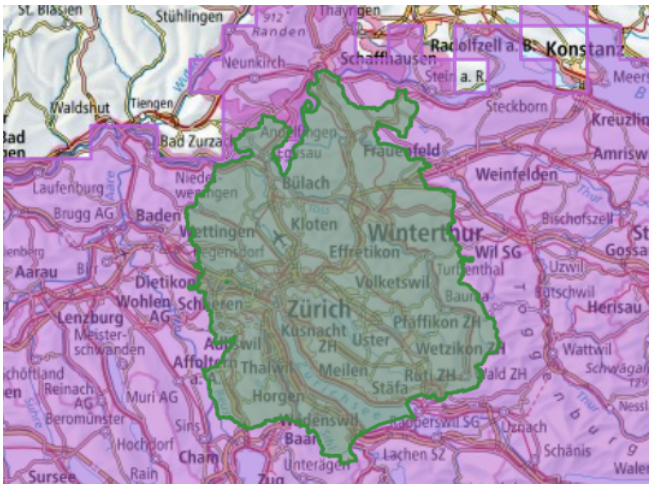


Figure 4.4: Location of study area

4.4 Evaluation Metrics

Considering that DSM terrain features are different from image features and also the problem at hand is a regression analysis, we choose various metrics to evaluate the model's performance. The metrics chosen for this research work are Root Mean Squared Error(RMSE), Mean Absolute Error(MAE), Median Absolute Error(MedAE), and Normalized Median Absolute Deviation(NMAD).

4.4.1 RMSE

RMSE is a commonly used statistical metric for evaluating the accuracy of predictions or models. It is used to assess the variations between observed and predicted results and offers a mechanism to express the overall error or deviation of a model's predictions from the actual data [27]. The RMSE is calculated as the square root of the mean of the squared differences between predicted and actual values. The formula for RMSE, given a dataset with 'n' data points/observations, is in Equation 4.1. Here y,x, and n represent ground truth, estimated values, and number of observations respectively. The model's predictions are more closely matched to the actual data points when the RMSE values are smaller. In other words, lower RMSE values denote more accurate model performance. In this research, the RMSE is expressed in meters. RMSE penalizes larger errors more heavily because of the squaring operation and it is sensitive to outliers since large errors contribute significantly.

$$RMSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2} \tag{4.1}$$

4.4.2 MAE

MAE is also another widely used metric for regression models and is mostly used along with RMSE to evaluate the model's accuracy. Similar to RMSE, MAE also measures the difference between predicted and actual values, but it takes an average of absolute differences [27]. This makes MAE less sensitive to outliers [49]. The calculation of MAE given a dataset with n data points is in Equation 4.2. MAE provides an average measure of how far off the model's predictions are from the actual data points. Smaller MAE values represent more accurate models, and they quantify the absolute magnitude of errors. Because MAE uses absolute differences, it does not disproportionately penalize huge errors as RMSE does. It gives equal weight to all errors, regardless of their size or direction (overestimation or underestimation). MAE doesn't provide information about the spread or variance of errors.

$$MAE = \sum_{i=1}^{D}|x_i - y_i| \tag{4.2}$$

4.4.3 MedAE

MedAE is a metric similar to MAE but instead of taking the mean of the absolute differences between predicted and actual values, MedAE takes the median of absolute differences [49]. This makes the metric even more resistant than MAE from outliers. The calculation of MAE given a dataset with n data points is in Equation 4.3. MedAE is particularly useful when dealing with datasets that may contain outliers or data points with very large errors. It is less sensitive to extreme values and provides a more accurate reflection of the central tendency of the errors.

$$MedAE = median(|x_i - y_i|) \tag{4.3}$$

4.4.4 NMAD

NMAD is a statistical metric used to measure the spread or dispersion of a prediction, similar to the MedAE. It is particularly useful when you want to understand the relative dispersion of data points, especially in the presence of outliers [49]. NMAD is obtained by calculating the median of the absolute difference between prediction and MedAR and dividing by a scaling factor. The scaling factor is typically a constant value chosen to make NMAD comparable or interpretable. Common choices for the scaling factor include values like 1.4826, which is used to make NMAD consistent with the standard deviation of a normally distributed dataset [49]. NMAD measures the spread or variability of data points relative to a scaling factor. NMAD is also robust to outliers because it relies on the median rather than the mean and uses absolute differences. The Equation 4.4 describes the metric.

$$NMAD = \frac{median(|x_i - MedAE|)}{scalingfactor} \tag{4.4}$$

4.5 Experimental setup

The computational setup used for this research work is from the DLR. Their cluster of GPUS contains various GPU cards but the experiments for this research utilized 4 GeForce GTX TITAN X GPUs with 16GB RAM and 12GB GPU memory. Ubuntu 20.4 is used as the OS for the Linux machine. All the experiments are tracked with respective metrics and training losses using the Weights&biases experimental tracking software. Pytorch has been used as deep deep-learning framework for this research.

5

Experiments & Results

This chapter discusses various experiments performed for this research work and their results.

5.1 Baseline model

As discussed in the methodology, this section will describe the experiments performed to establish a baseline deep learning model for DSM SR. The models being experimented with are D-SRGAN, Efficientnetv2, ESRGAN, Real-ESRGAN, and pix2pix(U-Net). Each model's network architecture and results are described in the following sections. Before going into detail about the experiments, there are a few training hyper-parameters that are the same across all the experiments for establishing a baseline model and also other experiments being performed in this research work. Training is performed using Adam optimizer with β values of 0.5 and 099 and a learning rate of 0.00001 for 150 epochs. The batch size used for training and validation is four, validation is performed every epoch and the best model is saved based on less MAE value. As mentioned in methodology Chapter 4, a Patch-based discriminator is used as a discriminator for these experiments unless specifically mentioned.

5.1.1 D-SRGAN

SRGAN [28] model is the first generative model used in image super-resolution tasks. D-SRGAN is the model used for DTM SR in [6], which is an inspiration with small changes in the design of generator architecture. The only difference between these two models is that the D-SRGAN generator uses the Tanh activation layer as the last layer, the number of initial feature layers used in the SRGAN generator is only 64 but D-SRGAN uses 128 and SRGAN uses 16 residual blocks but D-SRGAN only uses 10 residual blocks [28]. Apart from these changes in generator architecture, the SRGAN model uses the pre-trained VGG-19 network for the perceptual loss calculation but this is of very little help in the case of elevation model super-resolution. The model has two components Generator G and Discriminator D as any GAN model described in Chapter 4. The G network was designed to recover 4x downsampled images. This D-SRGAN network consists of a first layer with a convolutional layer followed by 10 residual blocks, 2 upsampling blocks, and a last convolutional layer to reduce feature maps to match the channel size to the input DSM channel size. Each residual block contains 2 convolutional layers with a 3*3 kernel, stride 1 and padding 1, parametric Relu as an activation function and two batch normalization layers with

skip connections. The upsampling blocks in this network use sub-pixel convolutional layers. Figure 5.1 illustrates the D-SRGAN generator architecture. Traditionally transposed convolutional layers were used for upsampling before SRGAN models but they cause the checkerboard artifacts [50], which is the reason why all the network designs from SRGAN in super-resolution use this sub-pixel convolutional layer for upsampling [28]. The combination of loss functions used for this model training are content loss (L1) and adversarial loss.

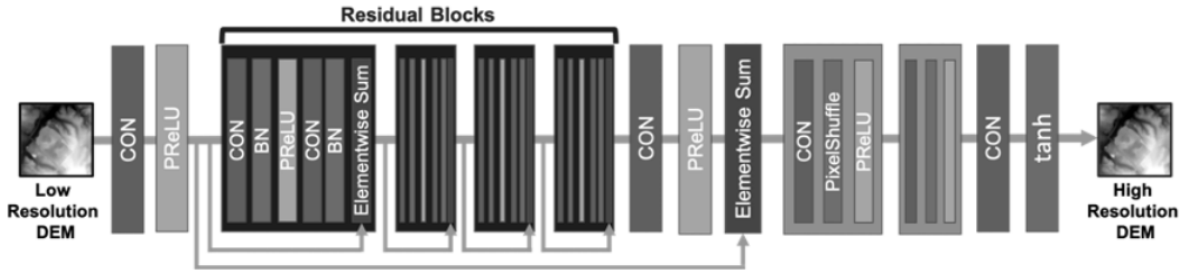


Figure 5.1: D-SRGAN generator architecture [6]. For experimenting purposes, the batch normalization layers and Tanh activation were removed. LR DSM was given as input, which passes through the residual layers for feature extraction, and later spatial dimensions were upsampled by upsampling blocks and finally produced a high-resolution DSM.

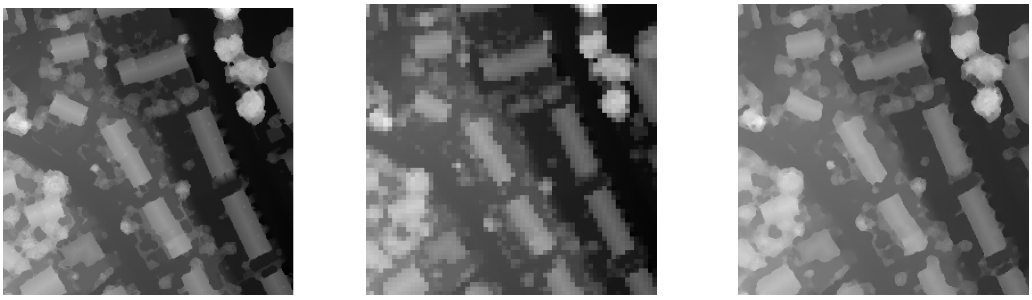
Apart from the actual design of D-SRGAN, small changes are made to the generator network for experimenting purposes. The changes are removing batch normalization layers and the Tanh activation function in the last layer. The dataset was normalized with standard deviation as mentioned in the dataset description in Chapter 4, due to which output after tanh activations (range between -1 and 1) are abnormal and adversarial losses are unstable with very high generative loss. Therefore, Tanh activation function was removed from the last layer of the network. The second change is about removing batch normalization layers present in the residual blocks. Related work in super-resolution shows that batch normalization(BN) creates artifacts and limits the generation ability [29]. Therefore, the experimental results will illustrate the performance of D-SRGAN with and without BN layers in Table 5.1.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
D-SRGAN_noact_nobn	1.065	0.434	0.118	0.232
D-SRGAN_noact	1.068	0.448	0.139	0.298

Table 5.1: D-SRGAN quantitative results

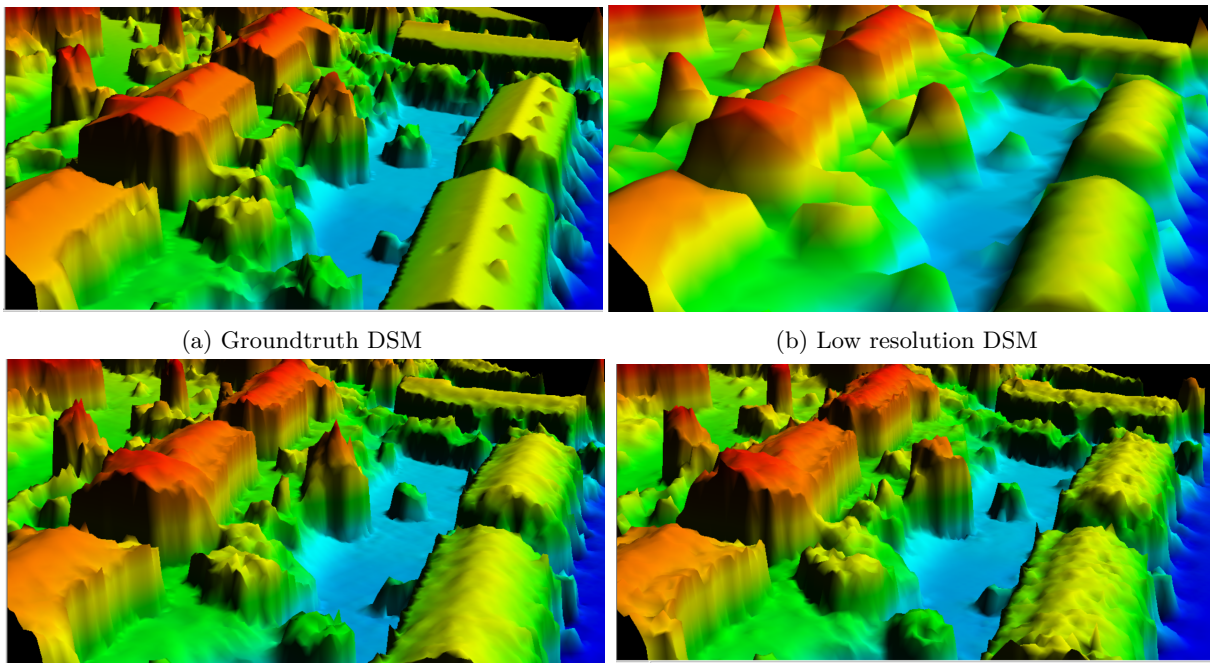
From Table 5.1, D-SRGAN without batch normalization layers gave slightly better numerical results compared to the model with batch normalization. This is also evident from the quantitative comparison of results between the two variants in Figure 5.3 as the model with batch normalization layers produced artifacts. Also, D-SRGAN results are better than the traditional Bicubic upsampling. From the 2D view

of DSM provided in Figure 5.2, the difference between SR DSM and HR ground-truth is hardly identifiable in the sense of fine details but these are visible in the 3D view in Figure 5.3.



(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

Figure 5.2: 2D comparison of D-SRGAN generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM (b) Low resolution DSM

(c) Super-resolved DSM (d) Super-resolved DSM with BN layers

Figure 5.3: 3D comparison of D-SRGAN generated DSM with groundtruth and input low resolution

5.1.2 ESRGAN

Enhanced SRGAN (ESRGAN) was proposed by authors in [29] as an improvement to the SRGAN model to improve the perceptual quality. The few modifications proposed by then in the generator network are the removal of BN layers and the replacement of residual blocks with Residual-in-Residual Dense

Blocks (RRDB) [29] as shown in Figure 5.4. ESRGAN model removed the BN layers based on the previous research stating that BN layers are bringing unpleasant artifacts [4]. RRDB blocks are employed to create a deeper and more complex architecture to extract better features in generating more realistic textures than SRGAN [29]. ESRGAN model is also designed to recover 4x downsampled data. Additionally, ESRGAN was a better-performing model in image super-resolution and state-of-the-art. Hence, the ESRGAN model was also experimented with to investigate whether ESRGAN has better performance than SRGAN in the case of DSM SR. In addition to the improved generator design, the ESRGAN model enhances the discriminator performance by using a Relativistic discriminator [29]. A relativistic discriminator seeks to predict the likelihood that a real image is comparatively more realistic than a false one, as opposed to the normal discriminator D in SRGAN, which estimates the likelihood that one input image is real and natural. The difference between the standard discriminator and relativistic discriminator is shown in Figure 5.5. Additionally, the relativistic discriminator is pixel-based and uses the spectral normalization layer instead of batch normalization. Spectral Normalization is a new normalization technique introduced by authors in [51] for "controlling the Lipschitz constant of the discriminator through the spectral norm of each layer". The main purpose of this normalization is "by not allowing the discriminator network to be overly sensitive to input changes, the generative adversarial network becomes more stable during training" [52]. Therefore, ESRGAN model training results with relativistic and patchGAN will be presented in this section to compare the effect of the relativistic discriminator.

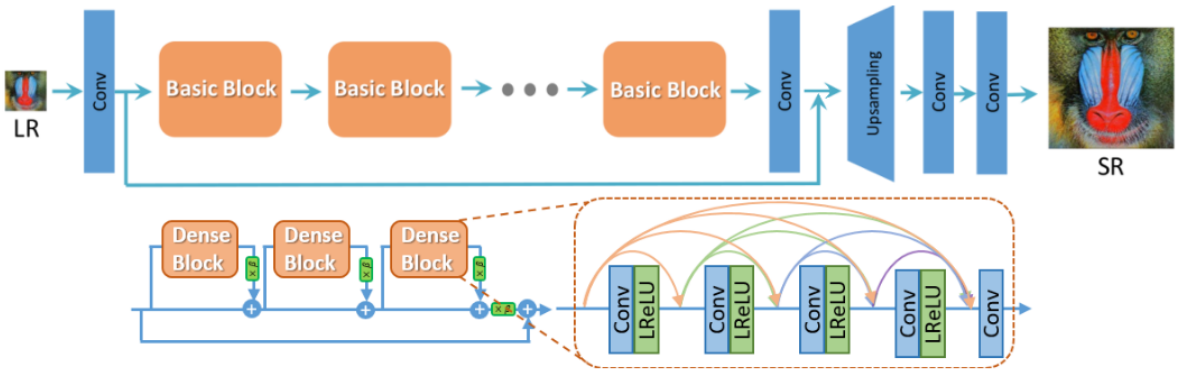


Figure 5.4: Architecture of ESRGAN generator. Input to the network is a low-resolution DSM and generates a 4x upsampled DSM. The model contains 23 basic blocks, each basic block contains 3 dense blocks and each dense block contains 5 convolutional layers, and 4 LeakyReLu activation functions with dense residual connections. Each convolutional layer uses a 3*3 kernel, stride 1, and padding 1. The output from 23 basic blocks is passed through the convolutional layer and concatenated with a long skip connection. latter passed through upsampling and 2 convolutional layers to produce an SR DSM

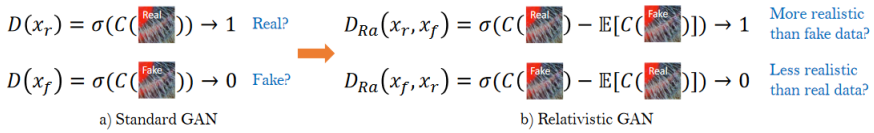


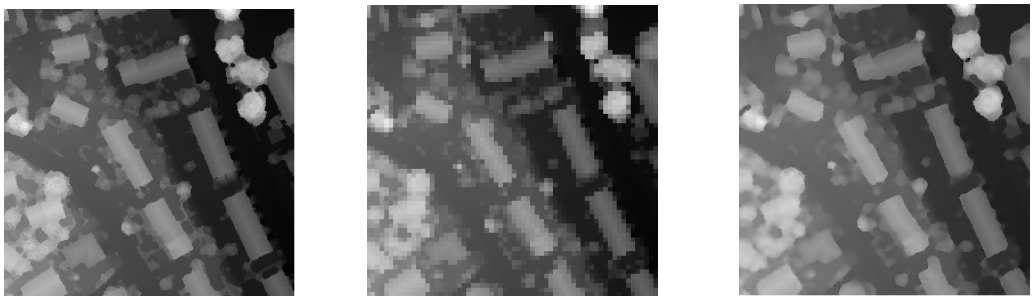
Figure 5.5: Difference between standard discriminator and relativistic discriminator [29]

Another variant of ESRGAN was introduced to super-resolve the images with multiple degradation known as Real-ESRGAN. The main objective of this Real-ESRGAN is to make a model capable of super-resolving the images with any real-world degradations possible. Therefore, the model was trained on synthetic data where the low-resolution data was generated by adding multiple degradations. Thus, the model knows how to super-resolve real-world degradations and produce realistic textures. Currently, this Real-ESRGAN is state-of-the-art in image super-resolution and comes under the category of blind super-resolution techniques. The generator used in Real-ESRGAN is the same as ESRGAN and uses a U-net style discriminator to handle these complex degradations. For the training of Real-ESRGAN, we have taken the pre-trained model available from their official Git Hub and finetuned our low-resolution DSM data as creating new synthetic data is not the objective for this thesis at this point in time. The training hyperparameters are the same as other experiments. The training results are illustrated in the Table 5.2.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
ESRGAN	1.087	0.467	0.164	0.292
ESRGAN_relatgan	1.023	0.425	0.126	0.523
RealEsrgan_unet_pretrained	1	0.41	0.119	0.237

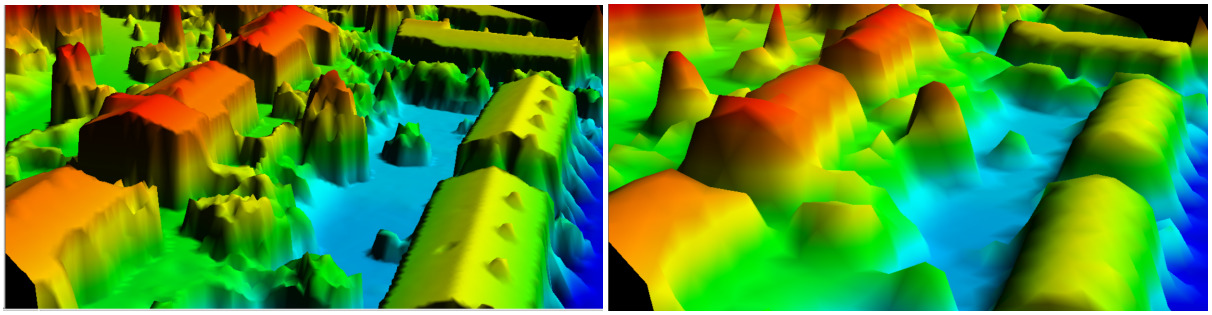
Table 5.2: ESRGAN model quantitative results using patch discriminator, relativistic discriminator, and Real-ESRGAN

From Table 5.2, we can see that the ESRGAN model performed better than traditional bicubic upsampling. On the other hand ESRGAN model with a relativistic discriminator produced slightly better numerical results in terms of RMSE and MAE metrics but qualitatively the patch discriminator produced visually better results without any artifacts. The reason for such numerical metrics is due to the use of spectral normalization which smoothens the data by smoothing the discriminator decision boundary [52]. But still has some small artifacts introduced by the relativistic discriminator as shown in Figure 5.7e. The quantitative results also stated that the finetuned Real-ESRGAN model resulted in better metrics than other ESRGAN models. From Figure 5.15h, the Real-ESRGAN model has fewer artifacts on the edges of the roofs of buildings when compared to the ESRGAN and ESRGAN with relativistic discriminator. Hence, from the experiments with ESRGAN, using a pre-trained model trained to handle various degradations helped in improving the performance of the model via transfer learning and converged the model smoothly.

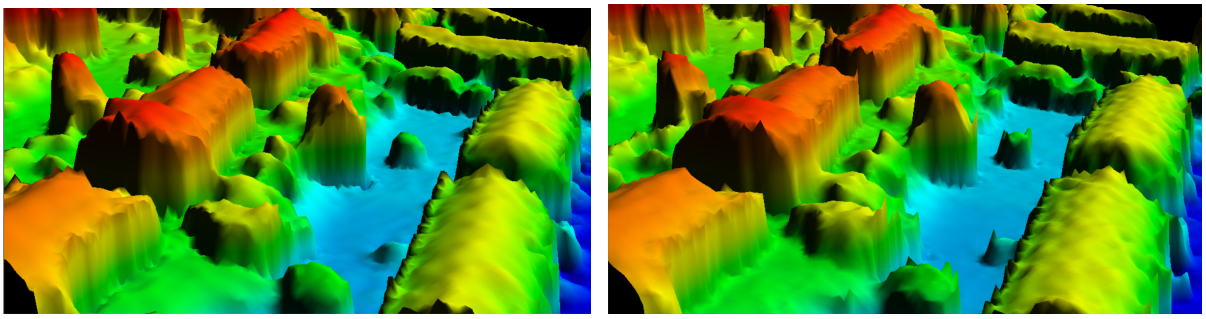


(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

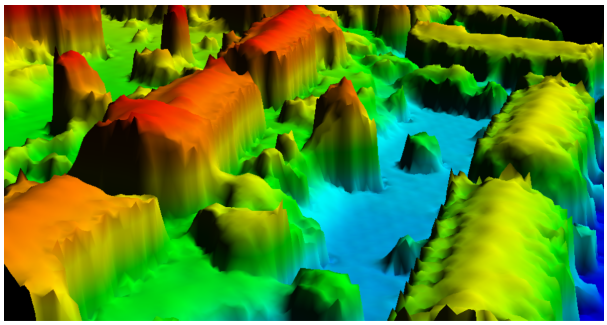
Figure 5.6: 2D comparison of ESRGAN generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM (b) Low resolution DSM



(c) ESRGAN Super-resolved DSM (d) Real-ESRGAN finetuned Super-resolved DSM



(e) ESRGAN Super-resolved DSM with relativistic discriminator

Figure 5.7: 3D comparison of ESRGAN, ESRGAN with relativistic discriminator and Real-ESRGAN models generated DSM with groundtruth and input low resolution

5.1.3 EfficientNetv2

EfficientNet is a family of convolutional neural network (CNN) architectures that are known for their efficiency in terms of computational resources and model size while achieving state-of-the-art performance on various computer vision tasks, such as image classification [53]. The key idea behind EfficientNet is to scale the model's depth, width, and resolution simultaneously to balance model size and performance. The improved version of EfficientNet was introduced by authors in [**?**] known as EfficientNetV2 to further improve the model's efficiency in computational resources. EfficientNetV2(Netv2) is the first time modified for super-resolution of DTM by authors in [54]. The main modification by authors to Netv2 is its main component MobileNetsV2 which is not designed for super-resolution tasks. Officially NetV2 model comes in 4 versions small, medium, large, and XL based on the amount of training parameters, and the authors used a small version for super resolution. The EfficientNetv2-small model was modified by us based on the description of the model from the paper as the code is not open-source. The EfficientNetv2 model code repository used in this research can be found here [55]. The small version of EfficinetNetv2 was utilized for the super-resolution task as other versions of the model downsample the spatial resolution of data in intermediate layers, which is not desired for this task. Similar to most of the super-resolution algorithms that came from SRGAN the spatial resolution is not changed until the last sub-pixel shuffle layers. This model upsamples 4x downsampled DSM. The architecture of this model for super-resolution can be shown in Figure 5.8. Similar to ESRGAN authors the authors of this model also choose to not use batch normalization layers present originally in the MobileNetv2 blocks.

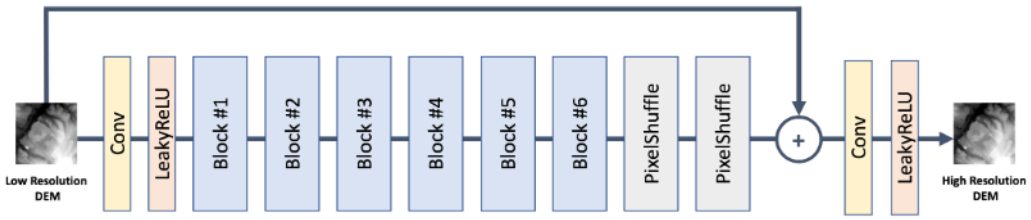


Figure 5.8: Architecture of the EfficientNetv2 used for super-resolution for DTM [54]. In our case, DSM low-resolution data is passed through the first convolution layer with 24 feature maps followed by passing into 6 MobileNetV2 blocks task and later passed through the sub-pixel convolutional layers for upsampling to reach the desired spatial resolution. An interpolated version of the low-resolution input data is concatenated with the output of the upsampling blocks at the end, and the concatenated output is then passed to the final convolutional layer, which produces the final output.

From the description from Figure 5.8, the 6 blocks in the architecture are MobileNetv2 blocks which contain inverted residual blocks (also sometimes called Fused-MBConv) that are introduced in cite [56]. Basically, "inverted residuals are a type of residual blocks that usually uses an inverted structure for efficiency reasons" [56]. The architecture of the Fused-MBCONv is shown in Figure 5.9. The SE block in MBConv is unchanged by authors for the super-resolution task. The authors for this Netv2 for DTM SR didn't train the model using adversarial loss only used L1 loss. As we are training GAN models for this

research, for this experiment, the EfficientNetv2 model is used as a generator and patch discriminator for adversarial training and the results from the model training are shown in Table **??**. From the results, we can observe that the traditional bicubic upsampling method produced better numerical results than this model. It is also evident from the qualitative results from Figure 5.15g.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
NetV2_noact_nobn	1.247	0.617	0.257	0.467

Table 5.3: EffcinentNetV2 quantitative results

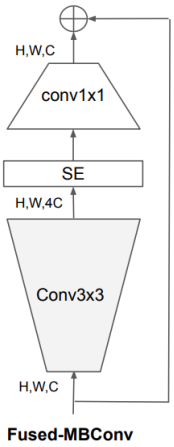
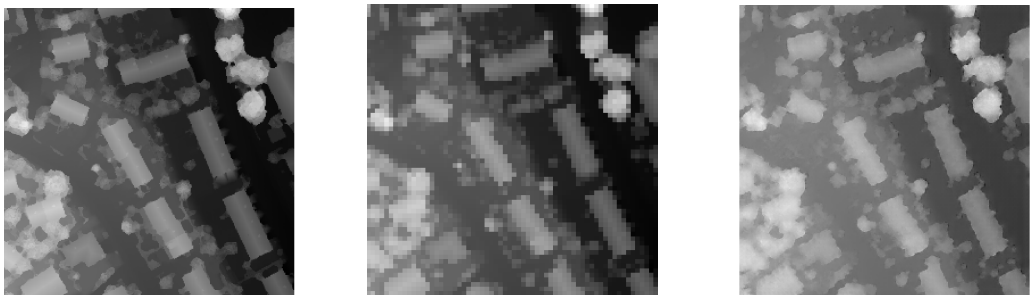
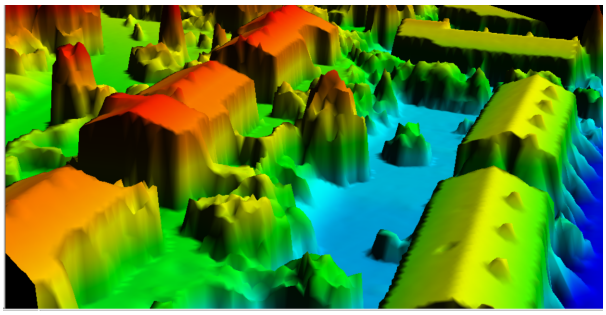


Figure 5.9: Fused-MBConv used in the efficientNetv2 [53]

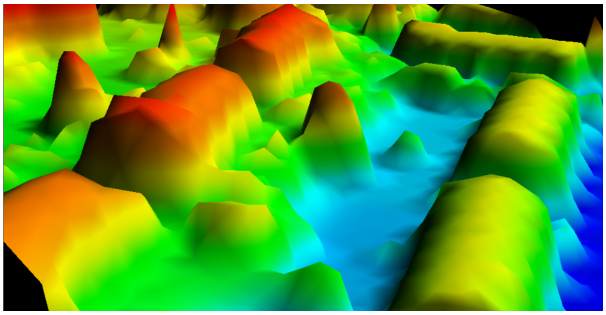


(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

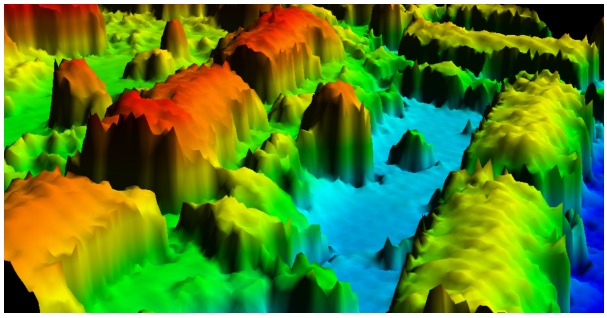
Figure 5.10: 2D comparison of EfficientNetv2 generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM



(b) Low resolution DSM



(c) Super-resolved DSM

Figure 5.11: 3D comparison of Netv2 generated DSM with groundtruth and input low resolution

5.1.4 U-Net(Pix2Pix)

Pix2Pix, short for "Image-to-Image Translation with Conditional Adversarial Networks," is a deep learning architecture that has been widely used for various image translation tasks. It was introduced by Phillip Isola in their 2016 paper [57]. Pix2Pix is particularly known for its ability to generate realistic images from input images and has found applications in tasks such as image colorization, image-to-image translation, and style transfer. This GAN model used a modified U-Net as a generator and patch discriminator for image translation tasks [57]. U-Net is an encoder-decoder model designed initially for semantic segmentation tasks and contains components such as encoder, decoder, bottleneck and skip connections. The Encoder is a contracting path that consists of convolutional and pooling layers that progressively reduce the spatial resolution while capturing contextual information. The decoder on another hand is an expansive path that concatenation operations to recover the spatial resolution. Skip connections from the encoder facilitate the flow of low-level and high-level features. The bottleneck is a bottleneck layer, typically comprising convolutional layers with a small receptive field. It serves as a bridge between the encoder and decoder, preserving spatial information. As the U-Net network input and output spatial dimensions should be of the same shape, the input low-resolution DSM is upsampled using bicubic and provided as input. As mentioned U-Net is basically an encoder-decoder architecture with skip connections between mirrored layers as shown in Figure 5.12.

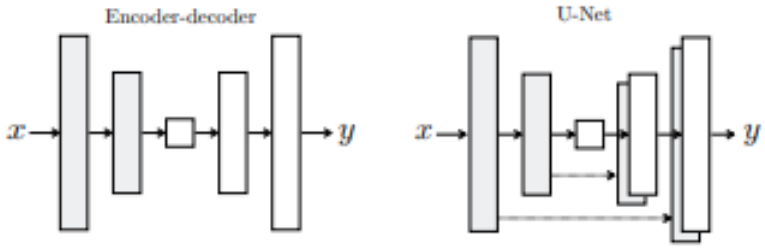
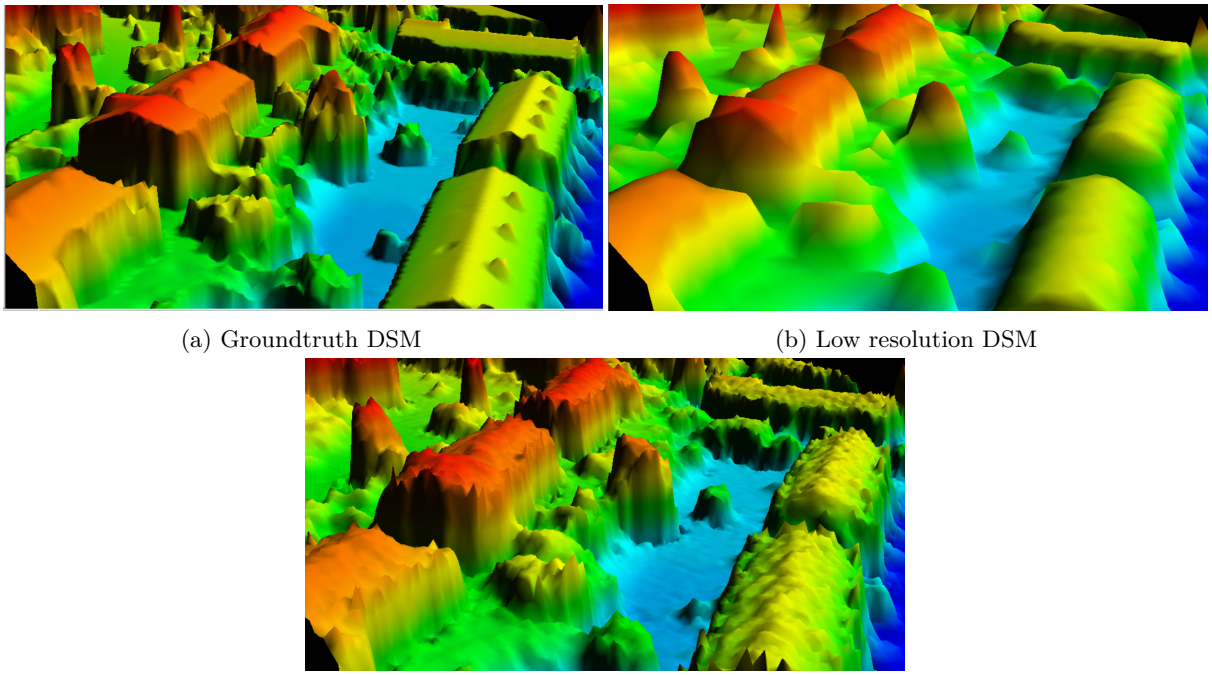


Figure 5.12: Difference between Encoder-Decoder architecture and U-Net [57]

From the results in Table 5.4, it is evident that the pix2pix model performance was close to bicubic, and also from Figure 5.18 the pix2pix model has more pointy artifacts on the surface when compared with high-resolution DSM and failed to generate an accurate DSM.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
unet_noact	1.046	0.499	0.194	0.411

Table 5.4: U-Net quantitative results

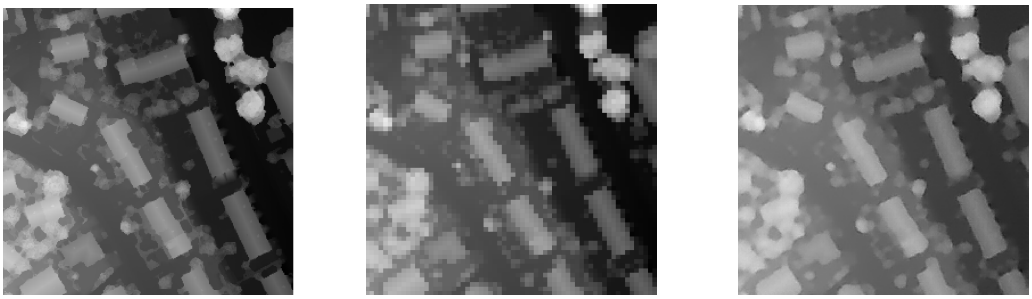


(a) Groundtruth DSM

(b) Low resolution DSM

(c) Super-resolved DSM

Figure 5.13: 3D comparison of Pix2ix model generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

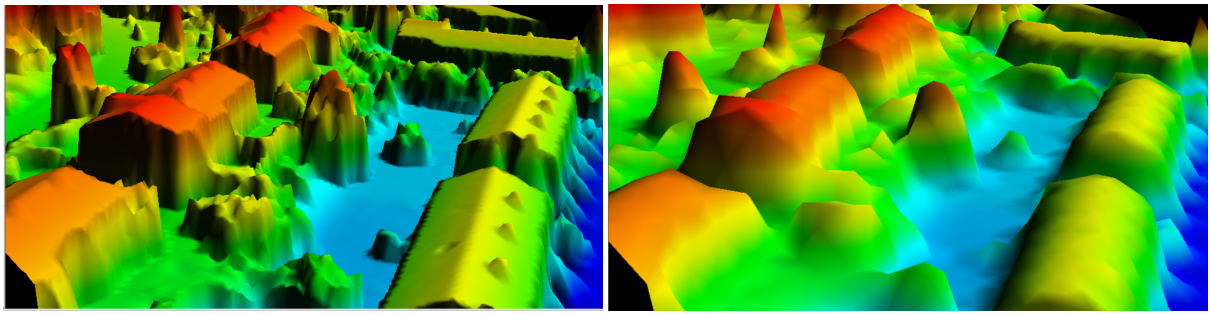
Figure 5.14: 2D comparison of Pix2Pix model generated DSM with groundtruth and input low resolution

5.1.5 Baseline model comparative Analysis

In the above experiments, we have tried five SOTA deep learning models which are used for image and/or DTM super-resolution for generating a super-resolved DSM. Table 5.5 illustrates the comparison of evaluation metrics for all the best training versions of the models discussed. Figure 5.15 provides the 3D view of the model's predictions. From all the results, we can infer that the D-SRGAN and finetuned Real-ESRGAN model provided better qualitative and quantitative results compared to the other 3 models and bicubic upsampling. The Real-ESRGAN achieved slightly better numerical results than SRGAN, which is 6.1 % less in RMSE and 5.52 % in MAE. However, very minute differences can be observed between these models qualitatively from Figures 5.15d and 5.15e. Therefore, considering the network complexity, computational power required, and 2-stage training of Real-ESRGAN, D-SRGAN was chosen as the baseline deep learning model. However, the best model still failed to generate a fine detailed urban DSM with structural high-frequency details from the comparative analysis of all the SOTA architectures.

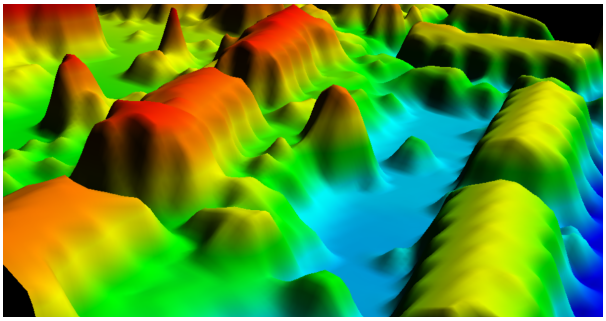
Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
D-SRGAN_noact_nobn	1.065	0.434	0.118	0.232
ESRGAN	1.087	0.467	0.164	0.292
RealEsrgan_unet_pretrained	1	0.41	0.119	0.237
NetV2_noact_nobn	1.247	0.617	0.257	0.467
unet_noact	1.046	0.499	0.194	0.411

Table 5.5: Baseline model qualitative analysis

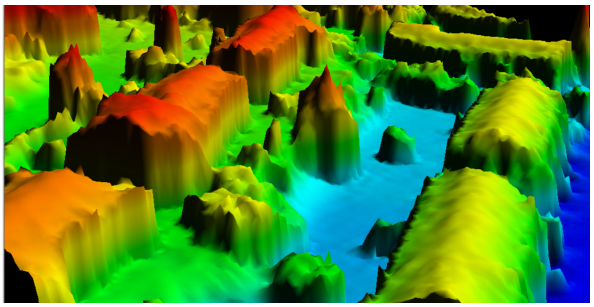


(a) Groundtruth DSM

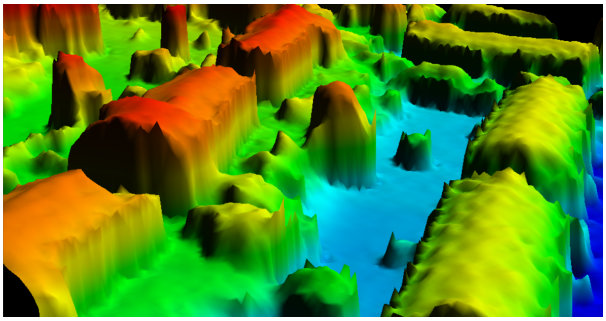
(b) Low resolution DSM



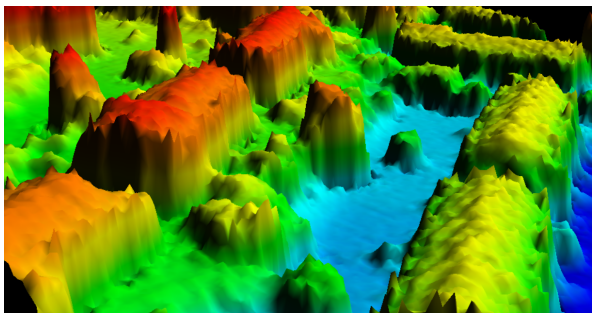
(c) Bicubic upsampled DSM



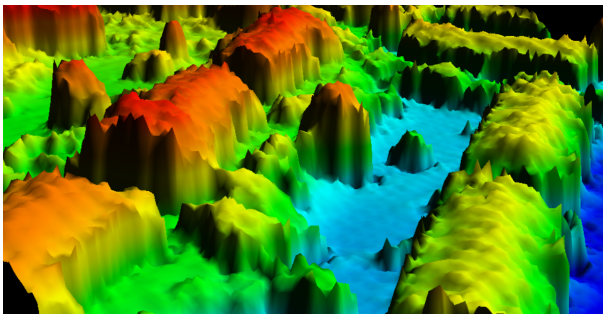
(d) Super-resolved DSM



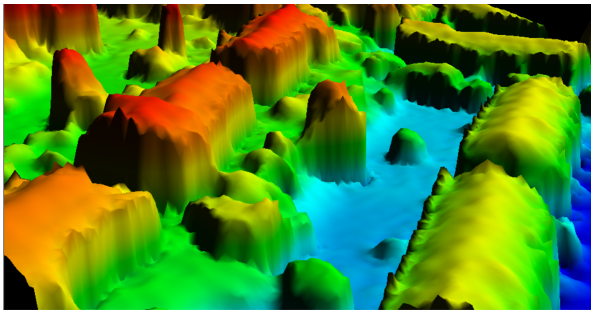
(e) Real-ESRGAN finetuned Super-resolved DSM



(f) Pix2Pix Super-resolved DSM



(g) EfficientNetv2 Super-resolved DSM



(h) ESRGAN Super-resolved DSM

Figure 5.15: Quantitative comparison of five deep learning models performance for super resolving the DSM

5.2 Attention mechanism

In recent times various deep learning models, particularly in the areas of natural language processing (NLP) and computer vision, now include attention processes as a basic aspect [9]. They enable models to concentrate on particular subsets of input data when creating predictions or output, which can greatly enhance their performance on tasks involving structured data. In other words, general "convolutional neural networks focus more on local information and ignore global features" [31]. An attention mechanism's primary function is to teach a model where to direct its "attention" within the incoming data. The most commonly used attention mechanisms in computer vision are Self-Attention and Channel-Attention [31]. Self-Attention is basically used in models like transformers, which allows elements within the same image to attend to each other, capturing dependencies and relationships. The model constantly picks and weights distinct input items, providing more weight to the elements that are relevant to the present job, rather than processing the full input image at once. On the other hand, Channel attention focuses on modeling the relationships between different channels (also known as feature maps) within a layer of a neural network [26]. Its primary goal is to emphasize or suppress specific channels based on their relevance to the task at hand. It addresses the challenge of capturing long-range dependencies in low-resolution images to generate high-resolution details effectively [26]. From the comparative analysis of establishing a baseline deep learning model for DSM, it is known that the SRGAN without batch normalization model is efficient in reconstructing DSM. Therefore, in order to improve further performance of this SRGAN model attention mechanism will be incorporated into the network and the experimental results will be discussed.

5.2.1 Channel Attention

Channel attention was first introduced by authors in [26] known as deep residual channel attention networks (RCAN) for image super-resolution. This model was designed to adaptively rescale channel-wise features by considering interdependencies among channels. The residual blocks in the RCAN network contain an attention mechanism by incorporating an average pooling layer, two convolutional layers, and a ReLu activation function. The average pooling enables the extraction of global spatial information and the channels are downsampled and upsampled by a ratio of K to extract the channel weights. By default or recommended ratio is 16 by the paper and we have used the same for this experiment. later sigmoid activation function was used as a gating mechanism for capturing the channel-wise dependencies from the aggregated information. The described channel attention block embedded along with two other convolutional layers and ReLu activation and a residual connection can be shown in Figure 5.16. This block is known as the residual channel attention block(ResCAB).

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
D-SRGAN_noact_nobn_rescab	1.00	0.467	0.174	0.349
D-SRGAN_noact_nobn	1.065	0.434	0.118	0.232

Table 5.6: Comparision of D-SRGAN with and without channel attention along with Bicubic

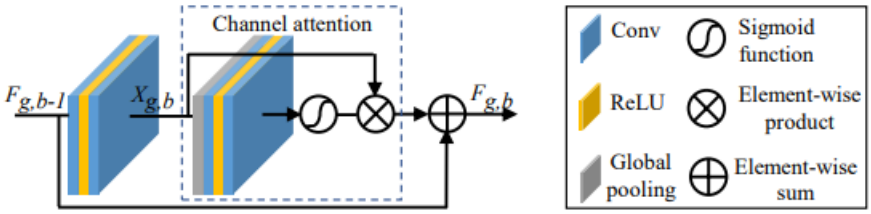
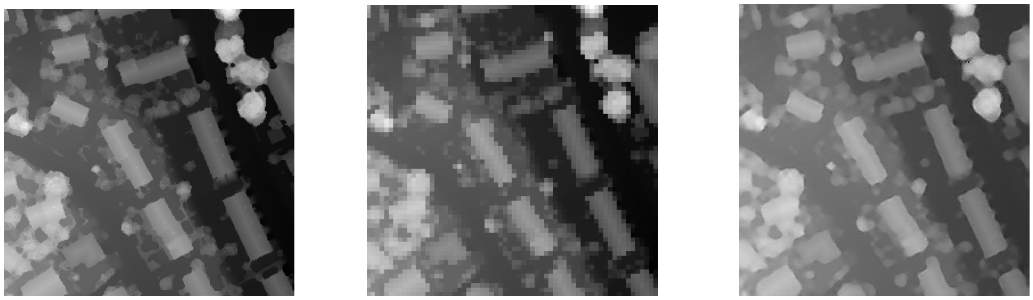


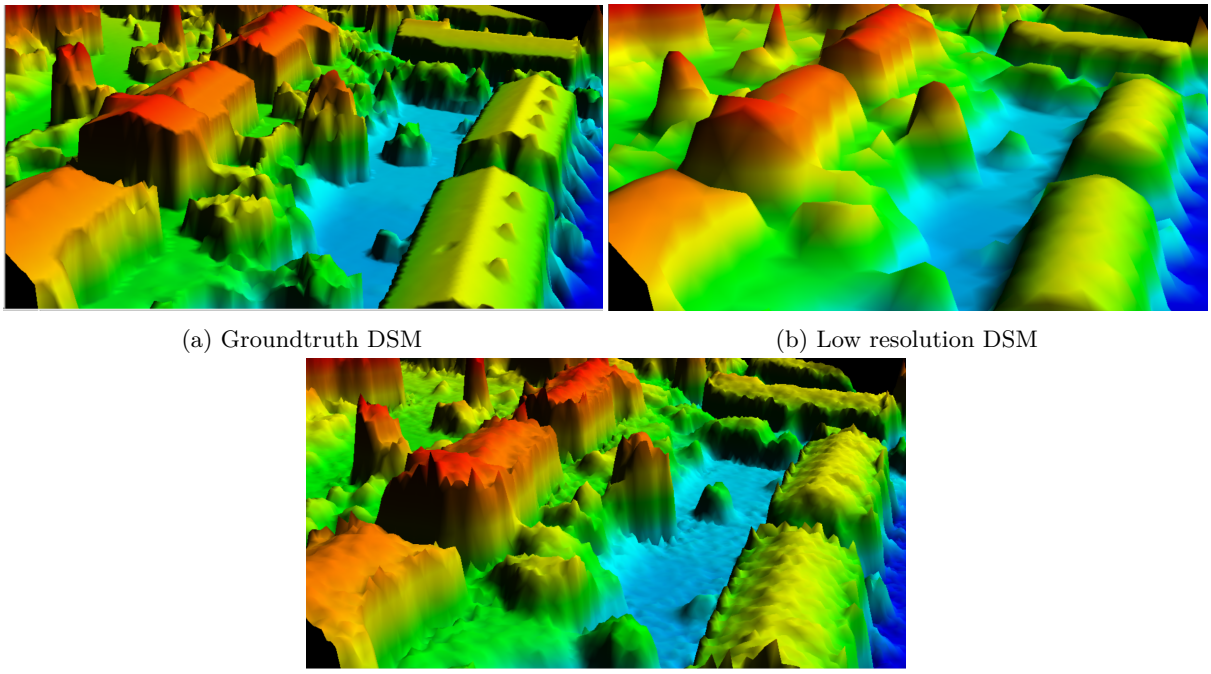
Figure 5.16: Residual channel attention blocks (ResCAB) [26]

As discussed, in this experiment SRGAN architecture will be modified by adding these ResCAB blocks to the generator network. 10 of these blocks will be added to the network behind the existing 10 residual blocks in the network. Next followed by the usual upsampling block and the last convolutional layer. The training of this GAN model is performed with usual hyper-parameters and the qualitative results for this experiment can be observed in Table 5.6. From the results, it is evident that channel attention has only slight improvement in terms of RMSE but in comparison to other metrics it didn't improve much of the performance of the SRGAN model. A detailed explanation of the reasons for such behavior will explained in discussions Section 5.6.



(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

Figure 5.17: 2D comparison of SRGAN-rescab model generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM

(b) Low resolution DSM

(c) Super-resolved DSM

Figure 5.18: 3D comparison of SRGAN-rescab model generated DSM with groundtruth and input low resolution

5.2.2 Self-Attention

Scaled dot-product attention is the standard and most commonly used self-attention and also the type used in transformer architecture [34]. Self-attention typically works on a set of input vectors, which are a grid of features in computer vision. For each element(pixels) in the grid Query, Key, and Value vectors are calculated. These vectors are linear transformations of input, computed using learned weight matrices. The Query and Key vectors are used to compute attention scores and determine the relationship between input elements, while the Value vectors represent the content of the elements. The attention scores are calculated using the dot product between the Query and Key vectors, followed by a scaling factor (usually the square root of the dimension of the Key vectors to prevent large values) [34]. The equation for the self-attention can be found in Equation 5.1. Here Q: Query vector for the current element, K: Key vector for all elements in the input sequence, and d_k: Dimension of the Key vectors. Later these attention scores are passed through the softmax function in order to obtain the normalized attention weights. The weighted sum of the Value vectors, using these attention weights, produces the final output for the current element as shown in Equation 5.2. In Equation 5.2 V is the Value vector for all elements in the input sequence. Additionally, Multi-head attention was also implemented where self-attention is computed multiple times in parallel each with different sets of learned weight matrices (Key, Query, and Value). These parallel attention computations are referred to as ”attention heads.” The outputs from multiple

attention heads are concatenated and linearly transformed to produce the final self-attention output.

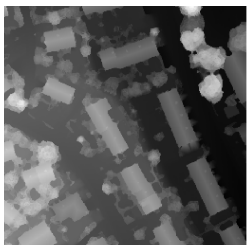
$$AttentionScore(Q, K) = \frac{(Q.K)}{sqrt(d_k)} \tag{5.1}$$

$$Attention(Q, K, V) = Softmax(AttentionScore(Q, K)) * V \tag{5.2}$$

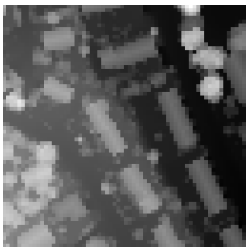
For this experiment, the single self-attention layer was added to SRGAN after the residual blocks and before the upsampling layers. Similarly, another experiment was tried with 8 multi-head attention layers to observe the effect of the deeper network. The experimental results from the training of these two SRGAN models along with bicubic upsampling can be found in Table 5.7. From the results, it can observed that by adding a single self-attention layer or multi-head attention the SRGAN model performance was not improved much compared to the SRGAN and besides that single-self-attention layer model performance was not even better than bicubic upsampling.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
srgan_noact_nobn	1.065	0.434	0.118	0.232
srgan_noact_nobn_multihead	1.053	0.482	0.174	0.392
srgan_noact_nobn_selfatten	1.129	0.56	0.207	0.561

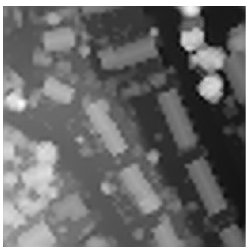
Table 5.7: Comparision of SRGAN with self-attention, multi-head attention and Bicubic



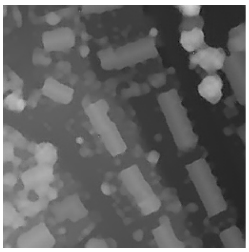
(a) Ground truth DSM



(b) Low resolution DSM

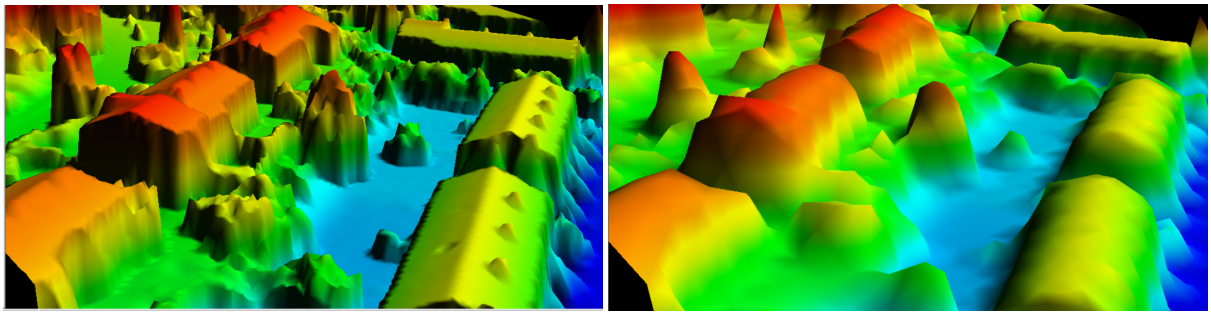


(c) Super-resolved DSM



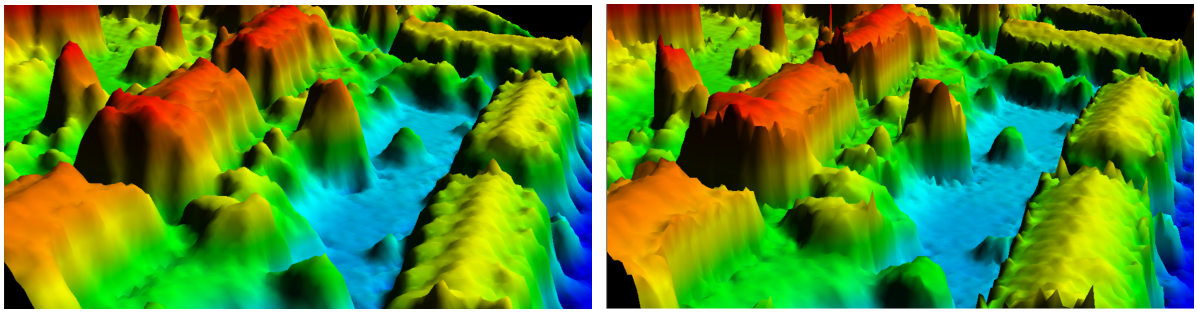
(d) Super-resolved DSM

Figure 5.19: 2D comparison of SRGAN self/multi-head attention model generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM

(b) Low resolution DSM



(c) Super-resolved DSM

(d) Super-resolved DSM

Figure 5.20: 3D comparison of SRGAN self/Multi-head attention model generated DSM with groundtruth and input low resolution

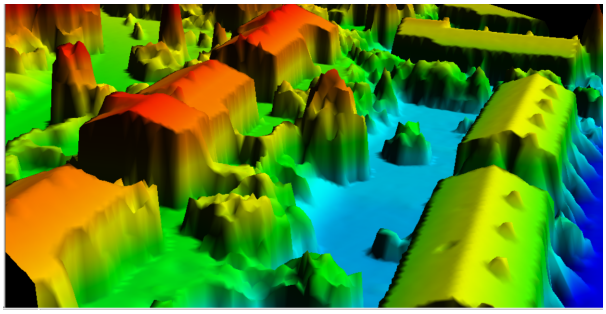
5.3 Co-learning

From the experiments discussed so far in previous sections, all the network architectures were designed to only perform 4x upsampling of the low-resolution DSM. Also from the results, we can infer that due to the lack of high-frequency features in low-resolution, DSM models are not able to retrieve the fine details of urban topography. Therefore, an experiment was performed by modifying the SRGAN for 2x upsampling by only using 1 subpixel convolution layer(upsampling layer). The low-resolution DSM samples were obtained by 2x downsampling the high-resolution DSM. Table 5.8 illustrates the performance of the SRGAN along with classical bicubic interpolation on the test set. From the results, we can observe that metrics for the SRGAN model for 2x upsampling are far better than classical bicubic upsampling with 34.51% in RMSE and 36.6% in MAE. whereas, in the case of 4x upsampling fromm the Table 5.1 the deep learning model SRGAN achieved only 2.82% in RMSE and 11.64% in MAE. These results indicate that model performance improved by having more features in the low-resolution DSM. Therefore, an approach to make 4X upsampling efficient by leveraging features from 2x upsampling has been identified and it is known as Co-learning.

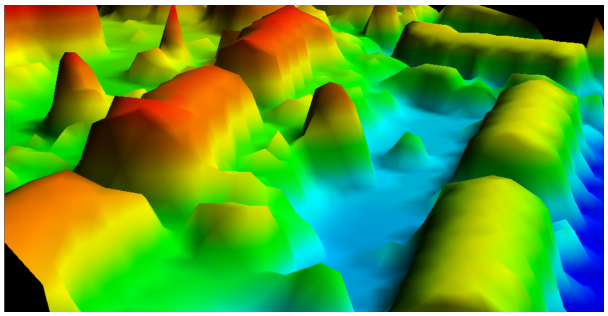
Co-learning is a new learning approach that utilizes collective knowledge and contributions to improve the efficiency, robustness, and performance of deep learning models. In [58] authors use co-learning for multi-modality collaborative learning to "adaptively exploit knowledge from another modality during the training phase with a soft connection, via a predefined loss function" [58]. "Instead of direct information

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic 2x	0.707	0.306	0.048	0.105
srgan_noact_nobn_2x	0.463	0.194	0.05	0.107
Bicubic	1.096	0.524	0.127	0.312
srgan_noact_nobn	1.065	0.434	0.118	0.232

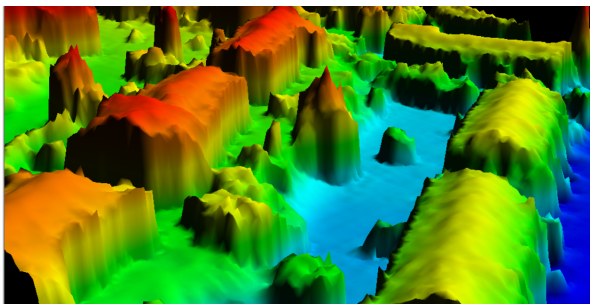
Table 5.8: Comaprision of SRGAN 2x with bicubic upsampling



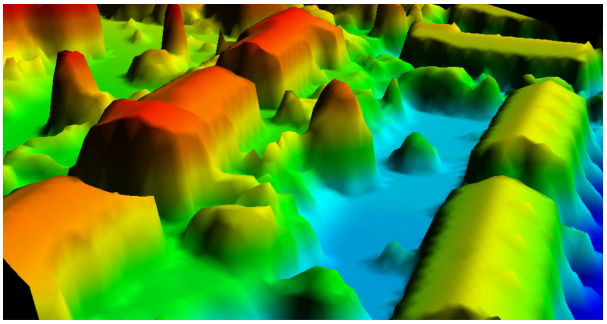
(a) Groundtruth DSM



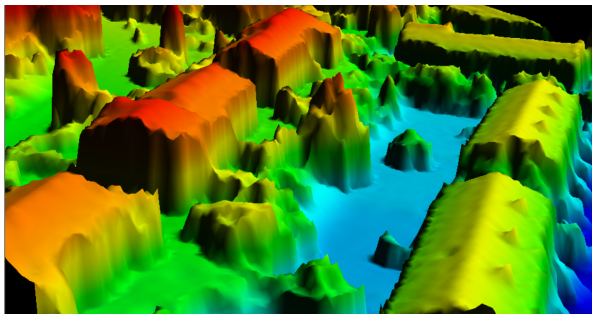
(b) Low resolution DSM



(c) Super-resolved DSM



(d) 2x Low resolution DSM



(e) 2x D-SRGAN Super-resolved DSM

Figure 5.21: Comparison of 4x and 2x D-SRGAN models output

fusion, this co-learning method is more flexible, as it is not mandatory to provide multimodality data in the test phase" [58]. As shown in Figure 5.22 authors use collaborative information from orthoimages and point clouds during training to improve each other by calculating the loss between the semantic maps

output from two models.

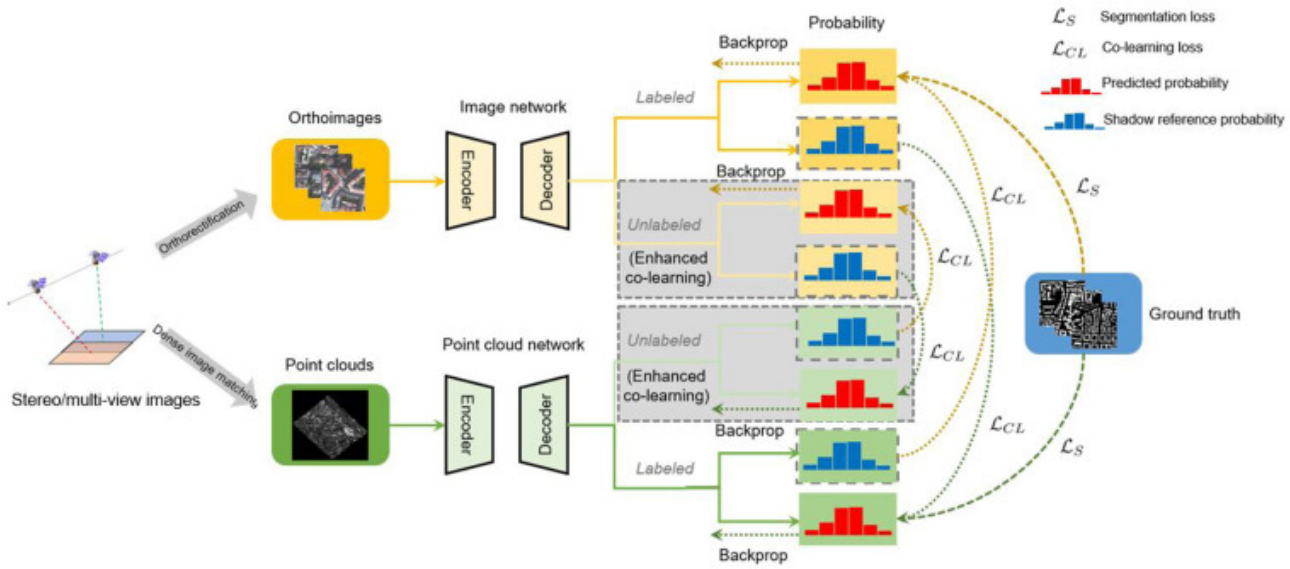


Figure 5.22: Co-learning architecture used for orthoimages and point clouds [**?**]

In a similar approach to the architecture presented in [58]. In this experiment, we train the co-learn model to minimize the co-learning loss between 2x output and 4x upsampled output to enable the 4x model to retrieve more fine details from the guidance of the 2x model. The co-learning loss used for this experiment is L1 loss. In Parallel, the L1 loss is also used for loss calculation between ground truth and respective model outputs. The proposed architecture for this experiment is shown in Figure 5.23. Additionally to just the content loss for the 4x model, adversarial loss was also calculated using the patch discriminator. The training hyper-parameters are the same as discussed in Section 5.1. The evaluation results from these experiments are illustrated in Table 5.9. From the results, we can infer that the co-learning architecture neither improved the model's feature extraction capability nor improved the resolution of the 4X output from guiding.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
srgan_noact_nobn	1.065	0.434	0.118	0.232
srgan_noact_nobn_colearn	1.066	0.497	0.187	0.418

Table 5.9: Comparision of SRGAN with SRGAN co-learn architecture

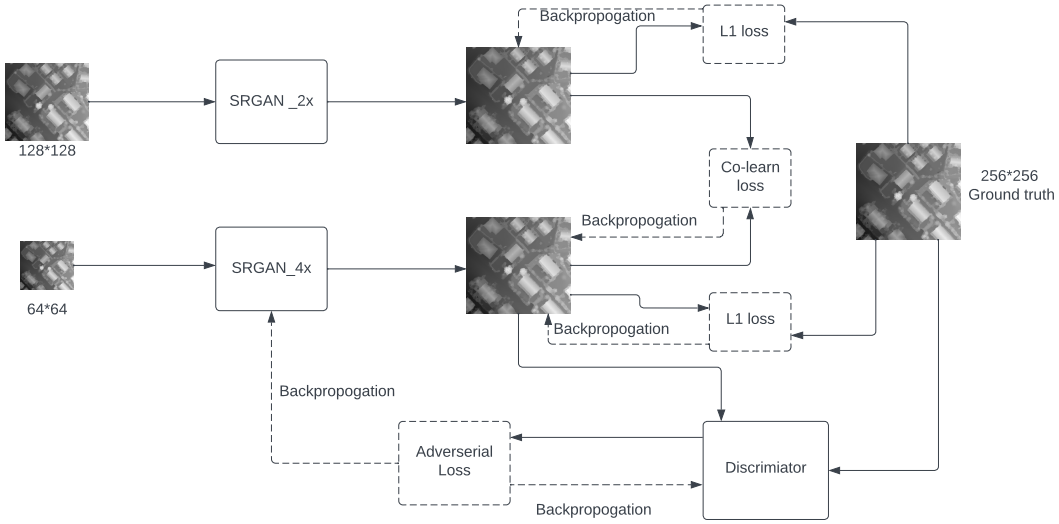
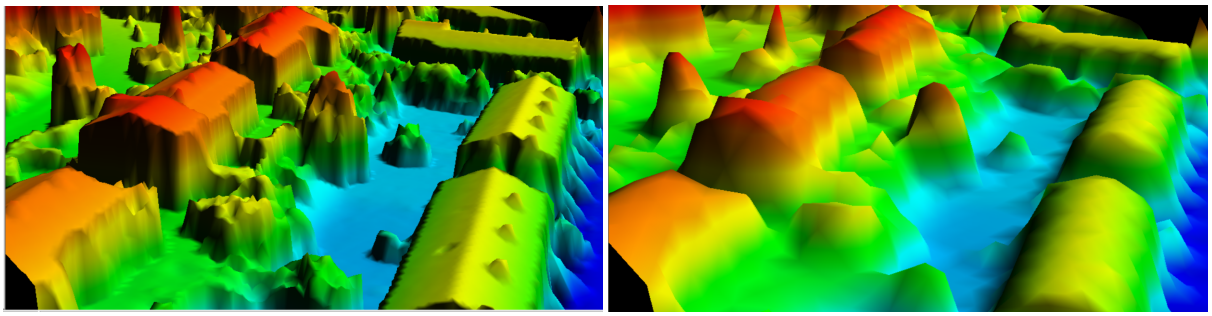
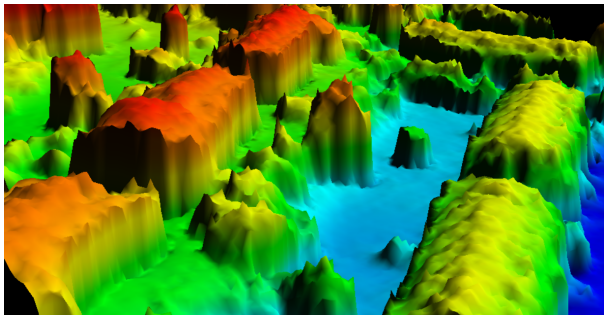


Figure 5.23: Co-learning architecture used for super resolution



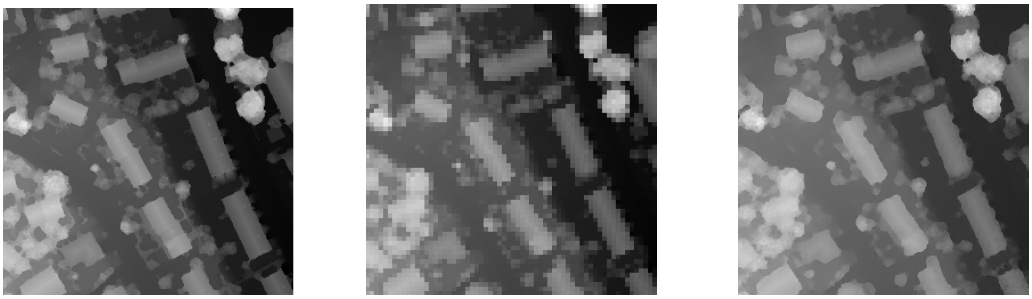
(a) Groundtruth DSM

(b) Low resolution DSM



(c) Super-resolved DSM

Figure 5.24: 3D comparison of SRGAN co-learning model generated DSM with groundtruth and input low resolution



(a) Groundtruth DSM (b) Low resolution DSM (c) Super-resolved DSM

Figure 5.25: 2D comparison of SRGAN co-learning model generated DSM with groundtruth and input low resolution

5.4 Encoder-SRGAN Model

Encoder-decoder architectures are a class of deep learning models that have demonstrated exceptional performance in a wide range of tasks, including machine translation, image captioning, speech recognition, and more. As discussed in section 5.1.4, the only difference between a basic Encoder-Decoder architecture and U-Net architecture is the skip connection and also discussed the working of components. In this particular experiment, a new Encoder-Decoder architecture was designed where the decoder is the SRGAN architecture. Encoder is a very small architecture that only contains two 3*3 kernel convolutional layers and 2 PReLu as activation functions, which deals with downsampling the 256*256 DSM to 64*64 with feature layers of 128 and then passed through 1*1 convolutional layer as a bottleneck and passed to the decoder architecture. The main idea for designing this architecture is to create a pre-trained model trained on high-resolution data. Later this pre-trained model can be finetuned on the low-resolution data to facilitate the transfer learning and faster convergence. The architecture of this model is shown in Figure 5.26.

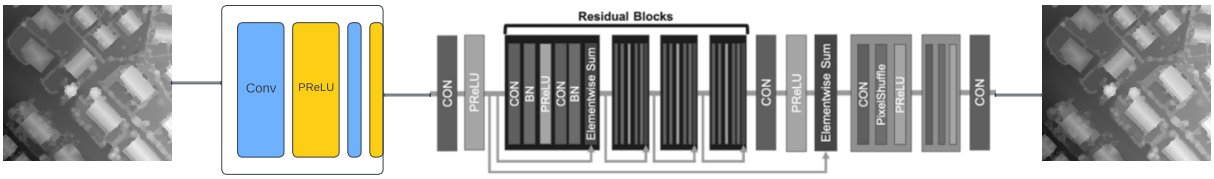


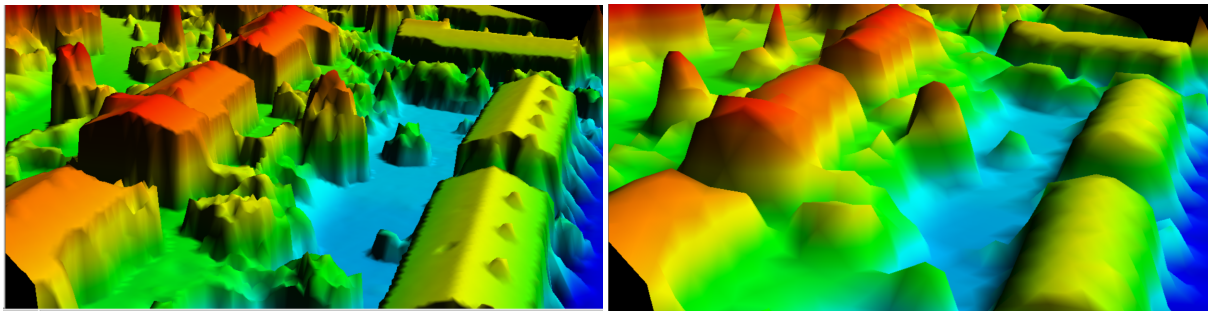
Figure 5.26: Encoder-SRGAN architecture. The encoder contains 2 3*3 kernel convolutional layers and 2 PReLu activation functions and a 1*1 kernel convolutional layer for bottleneck and SRGAN as decoder architecture. The input to the network is a bicubic upsampled low-resolution input (256*256) and the output of this model is a 256*256 super-resolved DSM

The training for this model is performed in a two-step process. The first step trains the model with high-resolution DSM using L1 loss without any adversarial training. In the second step, the pre-trained

model is used as a generator along with a patch discriminator to perform adversarial training by providing bicubic upsampled low-resolution DSM as input to the model. Other hyper-parameters used for this experiment are the same as we discussed in Section 5.1. The experimental results for this model training are illustrated in Table 5.26. From the results, we can infer that this design of the network and 2-step training didn't help in improving the model performance. Even though the model was pre-trained on high-resolution DSM, the model was not able to better generalize for the low-resolution DSM as input.

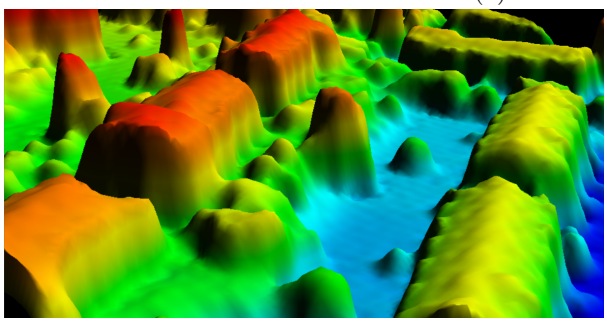
Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic	1.096	0.524	0.127	0.312
srgan_noact_nobn	1.065	0.434	0.118	0.232
enc_srgan_finetunned	1	0.453	0.134	0.264

Table 5.10: Comparision of Enc-SRGAN model performance with SRGAN and classical bicubic upsampling



(a) Groundtruth DSM

(b) Low resolution DSM



(c) Enc-SRGAN Super-resolved DSM

Figure 5.27: 3D comparison of Enc-SRGAN model generated DSM with groundtruth and input low resolution

5.5 DTM super-resolution

As mentioned in the proposal for this research work, the deep-learning models worked for DSM super-resolution will also be used for DTM super-resolution. Therefore, the D-SRGAN model is implemented for DTM super-resolution. Table 5.11 shows the numerical results. From the quantitative results, we can observe that the classical bicubic upsampling worked better than the deep learning model. Moreover, the

D-SRGAN model with multi-head attention proved to be efficient in super-resolving the DTM better than the baseline model.

Swiss dataset	**RMSE**	**MAE**	**MedAE**	**NMAD**
Bicubic_dtm	0.058	0.031	0.016	0.033
D-SRGAN_noact_nobn_dtm	0.0609	0.0362	0.022	0.047
D-SRGAN_noact_nobn_multihead_dtm	0.057	0.033	0.02	0.044
D-SRGAN_channel$_{a}$*tten_dtm*	0.063	0.039	0.026	0.05

Table 5.11: Comparision of D-SRGAN with channel attention, Multi-head attention, and Bicubic upsampling

5.6 Discussion

As we know, this research focused on developing a deep-learning based single image super-resolution algorithm for DSM super-resolution. The experiments in this chapter discussed various designs of architectures for this task. Initially, STOA algorithms for super-resolution in DTM and/or Image are used for establishing a baseline deep learning model. However, the baseline model generated DSM lacked fine-structured details present in an urban environment. Later the baseline model was modified in network architecture and learning to improve the generalization and reconstruction ability of the model.

- The results from all the experiments are tabulated in Table 5.12. From the quantitative results and the perceptual evaluation, we can infer that the attention mechanisms, co-learning, and encoder-decoder architectures didn't improve the baseline model performance.

- Therefore, in order to understand the reasons for such performance of the models, feature maps of the D-SRGAN model are illustrated for 2x and 4x sampling in Figure 5.28.

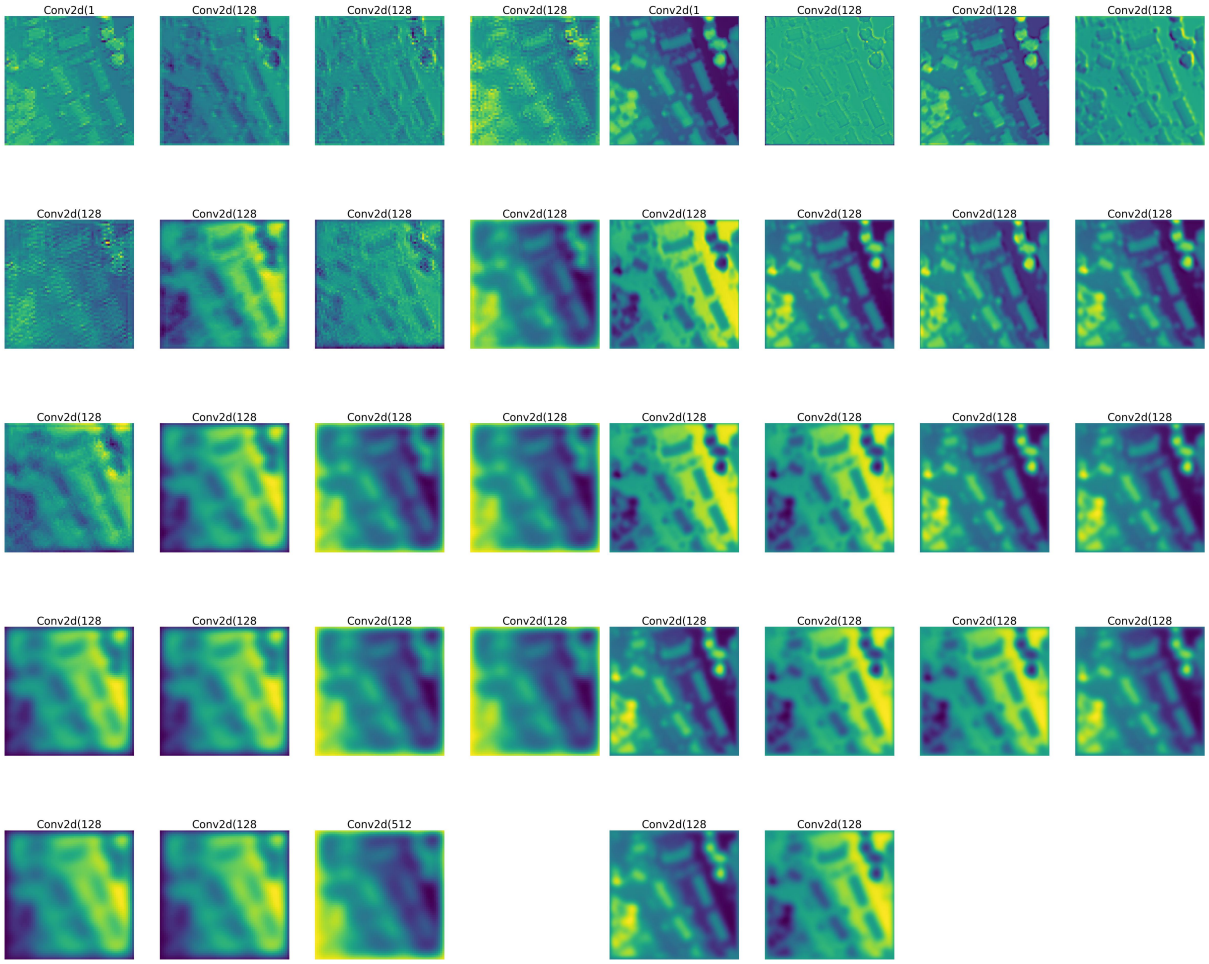
- From the feature maps, we can observe that the 2x downsampled input has better high-frequency features than the 4x input, and the model makes use of these features to reconstruct a high-resolution DSM. This can also be validated with the evaluation metrics tabulated in Table 5.8 and also from Figure 5.21.

- Therefore, this proves that due to the lack of fine high-frequency features in low-resolution input, the D-SRGAN model is only able to generate small high-frequency details resulting in missing fine structural details from the D-SRGAN generated DSM.

- The Enc-SRGAN model also failed to generate those fine details when provided a bicubic upsampled low-resolution input. Even though the model is pre-trained on the high-resolution dataset, the model failed to reconstruct a perceptually better DSM.

Swiss dataset	RMSE	MAE	MedAE	NMAD	DI
Bicubic	1.096	0.524	0.127	0.312	
D-SRGAN_noact_nobn	1.065	0.434	0.118	0.232	0.708
D-SRGAN_noact_nobn_resca	1.00	0.467	0.174	0.349	0.641
D-SRGAN_noact_nobn_multihead	1.053	0.482	0.174	0.392	0.575
D-SRGAN_noact_nobn_colearn	1.066	0.497	0.187	0.418	
Enc_SRGAN_finetunned	1	0.453	0.134	0.264	

Table 5.12: Quantitative results



(a) D-SRGAN 4x feature maps

(b) D-SRGAN 2x feature maps

Figure 5.28: Comparision of 4x and 2x D-SRGAN model feature maps

- The co-learning technique was mainly designed to influence the 4x model to generate DSM with fine details at least similar to that of the 2x model generated DSM. Even this way of training failed to improve the model's capacity to generate fine high-frequency details.

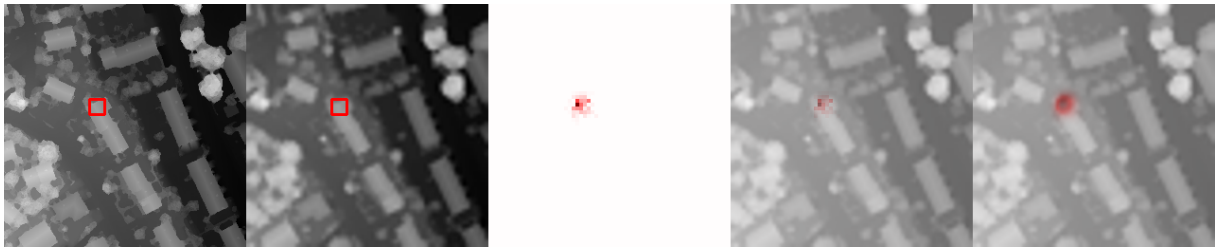


Figure 5.31: D-SRGAN model with Multi-head attention DI-0.575. Order of images: High-resolution DSM with the selected region, low-resolution DSM with the selected region, local attribution map result, local attribution map result on input, and informative area with the input image

- Therefore, from these co-learning and Enc-SRGAN models we can infer that mainly due to the lack of enough high-frequency features, the models are not able to improve the model's performance.

Figure 5.29: D-SRGAN model DI-0.708. Order of images: High-resolution DSM with the selected region, low-resolution DSM with the selected region, local attribution map result, local attribution map result on input, and informative area with the input image

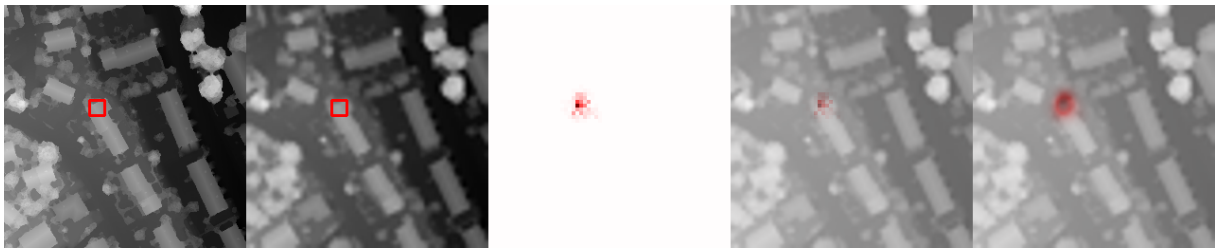


Figure 5.30: D-SRGAN model with residual channel attention DI-0.641. Order of images: High-resolution DSM with the selected region, low-resolution DSM with the selected region, local attribution map result, local attribution map result on input, and informative area with the input image

- Another way to interpret the model's performance in the case of super-resolution is by computing Local Attribution Maps(LAM) which were developed by authors in [59]. These attribute map "aims at finding the input pixels that strongly influence the SR results [59]. Additionally, a numerical metric is used along with attribution maps known as Diffusion Index (DI). The higher the DI value more pixels the model used for reconstruction.

- These attribution maps are "proposed to detect the existence of specific local features, such as edges and textures. We convert the problem of attribution into whether there exists edges/textures or not, instead of why these pixels have such intensities. In this manner, the attribution results are robust to the brightness changes" [59].

- Figure 5.29 illustrates the attribution maps for the DSM baseline D-SRGAN model, Figure 5.31 with multi-head attention and Figure 5.30 channel attention.

- From these attribution maps and respective DI values we can observe that the baseline model used more pixels for reconstruction than the baseline model with channel attention and multi-head attention. Between channel attention and multi-head model, the channel attention model utilized more pixels for reconstructing the region of interest.

- The DI value for these models supports the quantitative results for the models, where the model that used more pixels achieved better metric values. The respective comparison of metrics is shown in Table 5.12.

- Basically self/multi-head attention mechanism works by focusing its attention on long-range dependencies, which refer to the relationships or connections between distant pixels or image patches in a low-resolution input image that are crucial for generating a high-resolution output image. For instance, understanding that a texture or pattern repeats itself over a larger region of the image can help generate high-resolution details that are consistent with the overall structure.

- However, such feature patterns are not available in DSM data space due to their continuous nature, which is also evident from the attribute maps, where the pixels that influence the particular region of interest are not from distinct pixels or patches.

- Additionally, a similar argument is made by the authors of attribution maps, they observed that "even for the SR network with a large receptive field and good learning capacity, the area of noticing is still narrow. It indicates that these networks believe the semantics or features in a wider area have little help to the SR of the current patch" [59]. The examples considered by the authors of this argument can be observed in Figure 5.32.

- From Figure 5.32 we can observe that the images with regular stripes or grid structures across the whole image are better used by SR networks whereas, images with complex high-level semantics are hard to be detected by SR networks [59]. DSM Data used in this research also contains these complex high-level semantics.

- Therefore, from these observations, we can infer that the self-attention mechanisms failed to improve the performance of the baseline model due to a lack of long-range dependencies and complex high-level semantics.

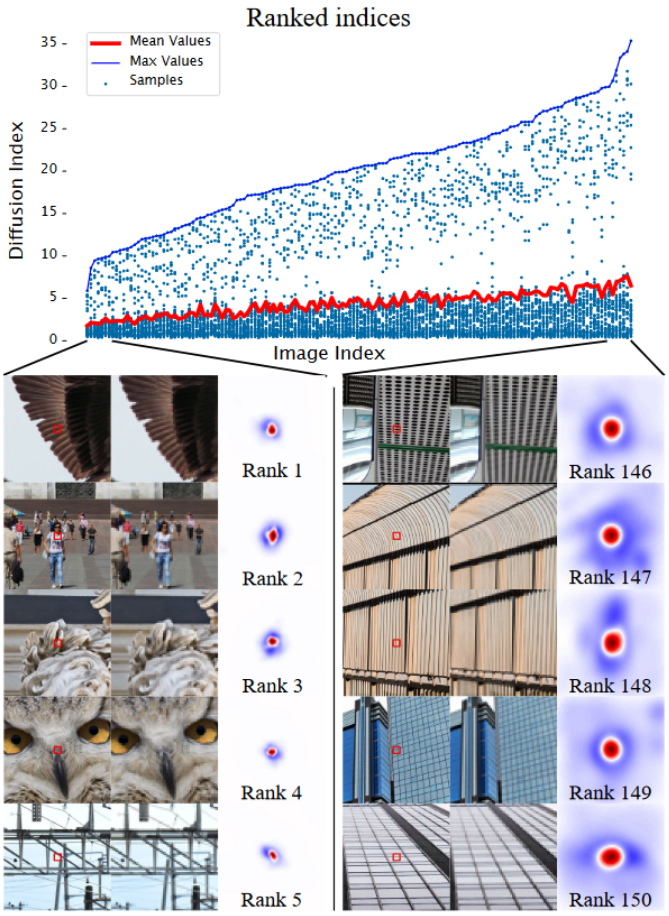


Figure 5.32: "Relationship of DI and image contents. The scatter plot shows the distribution of the DI values for different images and SR networks. The left bottom images with low-rank indices show the images with a narrow area of interest, and the right bottom images with high-rank indices show the images with a large area of interest." [59]

- From our experimental analysis we know that the Multi-head attention mechanism focused on long-term dependencies and such features are not available in DSM. So in our opinion, we can also conclude that it might not be possible for the basic transformer networks to work in reconstructing the DSM. Transformer architecture is based on multi-head attention blocks.

- On the other hand channel attention mechanism proved to be more efficient than self-attention and doesn't require long-range dependencies. This can be understood as the model's important channel from feature channels doesn't have the best features.

- Local attribution maps for the DTM are not much help in understanding the performance because authors from [59] stated that "the flat areas and simple edges are relatively easy to reconstruct in SR. The interpretation of these areas is of limited help in understanding SR networks." [59].

- However, the Multi-head attention mechanism improved the performance of the baseline model in the case of the DTM dataset.
- Therefore, from all these experimental results we can conclude that DSM feature space is different from general remote-sensing images feature space, which contains repeated patterns that help in effectively super-resolving the images with the focus of increasing the intensity value of pixels evenly across the image.
- Furthermore, from these experiential knowledge achieving an accurate super-resolved DSM with good structural details present in urban can only be possible if another source of information such as an image guides the super-resolution of DSM.

6

Conclusions

This research work focuses on the development of a deep-learning model for super-resolving a semi-urban DSM. This work intends to generate an accurate high-resolution DSM. From the literature review, DSM super-resolution remains an underexplored domain compared to image super-resolution. Most of the research in super-resolution is done in image data space and little research in DTM super-resolution, which is also inspired by image super-resolution models. Additionally, due to the complexity of the urban topography, the continuous nature of data, and the more high-frequency features present in DSM than DTM, the SISR developed for DTM may or may not reconstruct better high-resolution DSM. To bridge this gap, the study leverages state-of-the-art GAN-based deep learning algorithms for super-resolution such as D-SRGAN, ESRGAN, Real-ESRGAN, Pix2Pix(U-Net), and EfficientNetv2 to establish a baseline model for super-resolution of DSM. GAN models are known to be effective in generating sharp and realistic super-resolved output. Through comprehensive qualitative and quantitative analyses, D-SRGAN emerges as the baseline model, showcasing promising results. Subsequently, the research explores the development of additional models with D-SRGAN as a base to improve the baseline model performance. The proposed models do not significantly improve the performance of the baseline model and this can attributed to the unique characteristics of the DSM, lack of high-frequency features in 4x low-resolution DSM, and complex high-level semantics. Moreover, these models are not found to be feasible for enhancing resolution beyond a 4X scale. This thesis's conclusions contribute to our understanding of DSM super-resolution challenges and provide valuable insights for future research in this evolving field.

6.1 Contributions

The contributions of this research work are:

- **Literature review**: A detailed literature review is performed to identify state-of-the-art super-resolution architectures for images, DTM and DSM.
- **Implementation of algorithms**: Some of the deep learning models used in this research are implemented based on the description from the research papers.
- **Benchmarking super-resolution algorithmms**: Benchmarking is performed for DSM super-resolution using various state-of-the-art algorithms on the Swiss dataset.

6.2 Lessons learned

The major part of the research work is learning and understanding. The lessons learned are summarized as follows:

- **Understanding Elevation models**: Since this research is based on elevation models, it is essential to learn the data type, characteristics, and distinction between image and elevation model feature space.
- **Training Generative Adversarial Models**: Due to the possibility of instability, mode collapse, or failure to converge, GANs can be challenging to train. Stabilized the training of GANS using hyper-parameter tuning.

6.3 Future work

- Implementing diffusion architectures for the DSM super-resolution
- Implementing complex transformer architectures for the DSM super-resolution task.
- Multi-modality fusion of images and DSM during both training and testing could reconstruct the accurate structural details present in urban DSM. This can also achieve the task of super-resolving DSM at higher scales greater than 4x.

References

[1] Xu Lin, Qingqing Zhang, Hongyue Wang, Chaolong Yao, Changxin Chen, Lin Cheng, and Zhaoxiong Li. A dem super-resolution reconstruction network combining internal and external learning. *Remote Sensing*, 14:2181, 05 2022.

[2] Zhi Zheng, Yimin Luo, Zhang Yanfeng, Jun Wu, and Zhiyong Peng. A cnn-based subpixel level dsm generation approach via single image super-resolution. *Photogrammetric Engineering and Remote Sensing*, 85:765–775, 10 2019.

[3] DSM VS DTM. `https://shorturl.at/bsM38`. Accessed: 2023-04-14.

[4] Yifan Zhang and Wenhao Yu. Comparison of DEM super-resolution methods based on interpolation and neural networks. *Sensors*, 22(3):745, January 2022.

[5] Gary Priestnall, Jad Jaafar, and A. Duncan. Extracting urban features from lidar digital surface models. *Computers, Environment and Urban Systems*, 24:65–78, 03 2000.

[6] Bekir Z. Demiray, Muhammed Ali Sit, and Ibrahim Demir. D-SRGAN: DEM super-resolution with generative adversarial networks. *CoRR*, abs/2004.04788, 2020.

[7] Ling Jiang, Yang Hu, Xilin Xia, Qiuhua Liang, Andrea Soltoggio, and Syed Rezwan Kabir. A multi-scale mapping approach based on a deep learning CNN model for reconstructing high-resolution urban DEMs. *Water*, 12(5):1369, May 2020.

[8] Zixuan Chen, Xuewen Wang, Zekai Xu, and Hou Wenguang. Convolutional neural network based dem super resolution. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B3:247–250, 06 2016.

[9] Zhihao Wang, Jian Chen, and Steven Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 03 2020.

[10] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data, 2021.

[11] Peijuan Wang, Bulent Bayram, and Elif Sertel. A comprehensive review on deep learning based remote sensing image super-resolution methods. *Earth-Science Reviews*, 232:104110, 07 2022.

[12] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 12 2014.

[13] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. volume 9906, pages 391–407, 10 2016.

[14] Jiwon Kim, Jung Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. pages 1646–1654, 06 2016.

[15] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. *CoRR*, abs/1609.05158, 2016.

[16] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising, Aug 2016.

[17] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration, Apr 2017.

[18] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. *CoRR*, abs/1707.02921, 2017.

[19] Jianbo Jiao, Wei-Chih Tu, Shengfeng He, and Rynson Lau. Formresnet: Formatted residual learning for image restoration. pages 1034–1042, 07 2017.

[20] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. *CoRR*, abs/1511.04491, 2015.

[21] Donglai Jiao, Haiyang Lv, and Yang Peng. Super-resolution reconstruction of a digital elevation model based on a deep residual network. *Open Geosciences*, 12:1369–1382, 11 2020.

[22] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. *CoRR*, abs/1708.02209, 2017.

[23] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. *CoRR*, abs/1704.03915, 2017.

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, Los Alamitos, CA, USA, jul 2017. IEEE Computer Society.

[25] Jiu Xu, Yeongnam Chae, Björn Stenger, and Ankur Datta. Dense bynet: Residual dense network for image super resolution. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 71–75, 2018.

[26] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. *CoRR*, abs/1807.02758, 2018.

[27] Kai Li, Shenghao Yang, Runting Dong, Xiaoying Wang, and Jq Huang. A survey of single image super resolution reconstruction. *IET Image Processing*, 14, 07 2020.

[28] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. pages 105–114, 07 2017.

[29] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. ESRGAN: enhanced super-resolution generative adversarial networks. *CoRR*, abs/1809.00219, 2018.

[30] Seong-Jin Park, Hyeongseok Son, Sunghyun Cho, Ki-Sang Hong, and Seungyong Lee. *SRFeat: Single Image Super-Resolution with Feature Discrimination: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XVI*, pages 455–471. 09 2018.

[31] Xuan Wang, Jinglei Yi, Jian Guo, Yongchao Song, Jun Lyu, Jindong Xu, Weiqing Yan, Jindong Zhao, Qing Cai, and Haigen Min. A review of image super-resolution approaches based on deep learning and applications in remote sensing, Oct 2022.

[32] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. *CoRR*, abs/2006.04139, 2020.

[33] Xiangyu Chen, Xintao Wang, Jiantao Zhou, Yu Qiao, and Chao Dong. Activating more pixels in image super-resolution transformer, 2023.

[34] Z. Lu, J. Li, H. Liu, C. Huang, L. Zhang, and T. Zeng. Transformer for single image super-resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 456–465, Los Alamitos, CA, USA, jun 2022. IEEE Computer Society.

[35] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer, 2021.

[36] Sicheng Gao, Xuhui Liu, Bohan Zeng, Sheng Xu, Yanjing Li, Xiaoyan Luo, Jianzhuang Liu, Xiantong Zhen, and Baochang Zhang. Implicit diffusion models for continuous super-resolution, 2023.

[37] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting, 2023.

[38] Donglai Jiao, Dajiang Wang, Haiyang Lv, and Yang Peng. Super-resolution reconstruction of a digital elevation model based on a deep residual network. *Open Geosciences*, 12:1369–1382, 11 2020.

[39] Zekai Xu, Zixuan Chen, Weiwei Yi, Qiuling Gui, Hou Wenguang, and Mingyue Ding. Deep gradient prior network for dem super-resolution: Transfer learning from image to dem. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150:80–90, 04 2019.

[40] Dongjoe Shin and Steve Spittle. Logsrn: Deep super resolution network for digital elevation model *. pages 3060–3065, 10 2019.

[41] Zhou Annan, Yumin Chen, John Wilson, Heng Su, Zhexin Xiong, and Qishan Cheng. An enhanced double-filter deep residual neural network for generating super resolution dems. *Remote Sensing*, 13, 08 2021.

[42] Zherong Wu and Peifeng Ma. Esrgan-based dem super-resolution for enhanced slope deformation monitoring in lantau island of hong kong. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B3-2020:351–356, 08 2020.

[43] Di Zhu, Ximeng Cheng, Fan Zhang, Xin Yao, Yong Gao, and Yu Liu. Spatial interpolation using conditional generative adversarial neural networks. *International Journal of Geographical Information Science*, 34(4):735–758, April 2019.

[44] Xiaotong Deng, Weihua Hua, Xiuguo Liu, Siying Chen, Wen Zhang, and Jianchao Duan. D-srcagan : Dem super-resolution generative adversarial network. *IEEE Geoscience and Remote Sensing Letters*, pages 1–1, 2022.

[45] Github Repo. `https://github.com/Krishnateja244/DSM-super-resolution`. Accessed: 2023-10-10.

[46] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[47] Uğur Demir and Gozde Unal. Patch-based image inpainting with generative adversarial networks. 03 2018.

[48] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3d scene shape from a single image. *CoRR*, abs/2012.09365, 2020.

[49] Corinne Stucker and Konrad Schindler. ResDepth: A deep residual prior for 3d reconstruction from high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 183:560–580, jan 2022.

[50] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016.

[51] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[52] Spectral Normalization blog. `https://serp.ai/spectral-normalization/`. Accessed: 2023-05-25.

[53] Mingxing Tan and Quoc V. Le. Efficientnetv2: Smaller models and faster training. *CoRR*, abs/2104.00298, 2021.

[54] Bekir Demiray, Muhammed Sit, and Ibrahim Demir. Dem super-resolution with efficientnetv2. 09 2021.

[55] EfficientNetv2 Github Repo. `https://serp.ai/spectral-normalization/`. Accessed: 2023-05-25.

[56] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018.

[57] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.

[58] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. A co-learning method to utilize optical images and photogrammetric point clouds for building extraction. *Int. J. Appl. Earth Obs. Geoinformation*, 116:103165, 2023.

[59] Jinjin Gu and Chao Dong. Interpreting super-resolution networks with local attribution maps. *CoRR*, abs/2011.11036, 2020.