

Image Super Resolution Techniques Applied on Satellite Imagery

Chitipat Phentmune^{1,2} Hien Doan Thi^{2,3} Hoang Dieu Vu^{2,4}
Donghyun Ahn^{1,2} Hyunji Cha^{1,2} Sungwon Han^{1,2} Meeyoung Cha^{2,1}
¹KAIST, South Korea ²IBS, South Korea ³UET, Vietnam ⁴HUST, Vietnam

Abstract

This paper proposes an idea to apply deep learning models that generate super-resolution (SR) images for satellite pictures. We newly define the task by exploiting images of different zoom levels and demonstrate the results of implementing several recent models using over 12,080 samples. These models use satellite images at zoom level 16 and produce high-resolution images at zoom level 17 (i.e., four times the area of zoom level 16). We find that EDSR and RDN show significant improvement over Bicubic interpolation for images of urban built-up, but generate blurred output for other kinds of landcover such as vegetation. GAN-based methods create sharper images with details, although they achieve low PSNR scores. We confirm that different SR models are suitable for handling each landcover type on satellite imagery.

1. Introduction and Background

Recent studies have demonstrated how daytime satellite imagery can be used to extract meaningful economic and developmental insights over large geographical areas. As the zoom level increases by a unit, its resolution increases by two times, and the size of data that needs processing increases by four times (2×2). High-resolution images, however, are costly to obtain and often restricted in their usage, limiting their potential applications. Furthermore, as existing satellites wear out, the quality of images taken by those satellites tend to degrade. Hence the ability to reconstruct a high-resolution (HR) satellite image from a low-resolution (LR) one — *Super-Resolution (SR) technique* — could benefit the industry tremendously.

Some approaches use residual learning to connect the LR and HR images or between layers such as EDSR [3], yet these models suffer from vanishing and exploding gradients. Other methods like RDN [4] utilize DenseNet to mitigate gradient vanishing and handle the association of residual learning and dense block. Some approaches adopt generative adversarial networks (GAN). For instance, DeblurGAN [1] uses a perceptual loss to capture textual details. However, GAN-based models produce lower PSNR (peak signal to noise ratio), and they often introduce undesired artifacts under deeper networks. Models like EDSR [3] adopt the GAN loss [2] yet exploits the trade-offs between the PSNR value and realistic textual details.

Motivated by advances in SR techniques, our work newly defines the problem of constructing super-resolution land covers of broad geographic surfaces based on low-resolution images by exploiting the zoom-level structure in satellite imagery. We release a new dataset `SI_RAW` that includes 12,080 pairs of low and high-resolution satellite images collected from ArcGIS.¹ Our data comprise landcover images taken at both zoom level 16 (i.e., low-resolution) and zoom level 17 (i.e., high-resolution). We implemented and tested the following models: EDSR, DeblurGAN, RDN, and EDSR&GAN.

2. Methods and Findings

The experiment was conducted with a scaling factor of 2 at zoom level 16 and 17, which are reasonable to examine landcovers (e.g., water, vegetation, barren land, built-up). From the data, we discarded all cases where the LR and HR pictures were taken at different time periods and thereby increases the chance of observing differences in landcover, such as a new buildup structure. To ensure data quality, we utilized only those pairs that had gaps in PSNR scores more than 24. This process discarded 36% of LR-HR image pairs.

Overall, the EDSR and RDN models showed significant improvement over Bicubic interpolation in terms of the PSNR and SSIM scores (see Table 1). The PSNR scores were 30.82, 31.30, and 31.29 for Bicubic, EDSR, and RDN models, respectively. Visually, sample images of an urban built-up in Fig. 1 demonstrate that EDSR and RDN produce sharper and

¹We will release the implementation codes for four SR models via GitHub.

