MAPEANDO O INVISÍVEL: EXPLORANDO SUPER-RESOLUÇÃO PARA SEGMENTAÇÃO SEMÂNTICA EM IMAGENS DE BAIXA RESOLUÇÃO

MATHEUS BARROS PEREIRA

MAPEANDO O INVISÍVEL: EXPLORANDO SUPER-RESOLUÇÃO PARA SEGMENTAÇÃO SEMÂNTICA EM IMAGENS DE BAIXA RESOLUÇÃO

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Jefersson Alex dos Santos

Belo Horizonte
Novembro de 2019

MATHEUS BARROS PEREIRA

MAPPING THE UNSEEN: EXPLOITING SUPER-RESOLUTION FOR SEMANTIC SEGMENTATION IN LOW-RESOLUTION IMAGES

Dissertation presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: Jefersson Alex dos Santos

Belo Horizonte
November 2019

Ficha catalográfica elaborada pela bibliotecária Belkiz Inez Rezende Costa CRB 6ª Região nº 1510

Pereira, Matheus Barros.

P436m Mapping the unseen: exploiting super-resolution for semantic segmentation in low-resolution images / Matheus Barros Pereira. — Belo Horizonte, 2019.
xxiv, 64 f.: il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientador: Jefersson Alex dos Santos.

1. Computação – Teses. 2. Visão por computador – Teses. 3. Super-resolução – Teses. 4. Sensoriamento remoto – Teses. 5. Segmentação semântica – Teses. I. Orientador. II. Título.

CDU 519.6*82.10(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Mapping the Unseen: Exploiting Super-Resolution for Semantic Segmentation in Low-Resolution Images

MATHEUS BARROS PEREIRA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

Jefersson A. dos Santos
PROF. JEFERSSON ALEX DOS SANTOS - Orientador
Departamento de Ciência da Computação - UFMG

George L.M. Teodoro
PROF. GEORGE LUIZ MEDEIROS TEODORO
Departamento de Ciência da Computação - UFMG

Saúde
PROF. ANDRÉ VITAL SAÚDE
Departamento de Ciência da Computação - UFLA

Wesley N. Gonçalves
PROF. WESLEY NUNES GONÇALVES
Faculdade de Engenharias, Arquitetura e Urbanismo e Geografia - UFMS

Belo Horizonte, 11 de Novembro de 2019.

Aos meus pais, Fábio e Ângela.

Acknowledgments

Agradeço, primeiramente, aos meus pais, Fábio e Ângela, que me apoiaram incondicionalmente e deram todo o suporte que eu precisava a todo momento. O carinho e amor que recebo de vocês sempre foi essencial, mas durante esse período do mestrado foram mais importantes ainda. Às minhas irmãs, Ariana e Érica, por todo o amor e suporte que também recebi de vocês.

À minha namorada, Mariela, por todo o amor, carinho e companheirismo. Este é o período em que mais estivemos juntos até então, e sua companhia sempre foi extremamente importante para mim. Sei que posso contar com você para tudo!

Ao restante da minha família que de alguma forma também me apoiou e ofereceu suporte em diversas situações.

Agradeço ao meu orientador, Jefersson, por todos os ensinamentos, pela confiança e paciência. Agradeço também a todos os amigos do PATREO que fizeram esse tempo no mestrado ser muito mais fácil, seja me ajudando ou com o bom-humor de sempre que fazem os dias serem mais tranquilos aqui.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), que financiou este trabalho.

Por último, agradeço aos senhores Shinoda, Bennington, Bourdon, Delson, Hahn e Farrell. O trabalho de vocês sempre me motivou, inspirou e acompanhou desde a infância.

"Who cares if one more light goes out in the sky of a million stars? Well, I do"
(Linkin Park, One More Light)

Resumo

Imagens aéreas de alta resolução são desejáveis para a maior parte das aplicações de sensoriamento remoto baseadas em algoritmos profundos. Esse tipo de dado, contudo, nem sempre é acessível. Por outro lado, imagens de sensoriamento remoto de baixa/média resolução, como as dos satélites LANDSAT e MODIS, são facilmente encontradas em repositórios públicos abertos e, portanto, são usadas em diversos estudos. O problema é que a quantidade de informação espacial comprimida em um único pixel em uma representação de baixa resolução pode comprometer algoritmos de reconhecimento de padrão. Assim, o uso de dados de baixa resolução para a criação automática de mapas temáticos é muito restrito, dado que a maioria das abordagens baseadas em algoritmos profundos para segmentação semântica (ou rotulação densa) são adequadas apenas para dados subdecimais. Super-resolução é um problema clássico de visão computacional que busca restaurar a qualidade de imagens de baixa resolução. No presente trabalho, foram desenvolvidos dois arcabouços que têm como objetivo avaliar a efetividade de super-resolução baseada em algoritmos profundos na segmentação semântica de imagens de sensoriamento remoto de baixa resolução. Visa-se avaliar quão efetivo é a super-resolução em diferentes níveis de degradação, como se compara com interpolação bicúbica não-supervisionada e se é capaz de reconstruir objetos pequenos e, consequentemente, contribuir para o melhoramento da segmentação semântica. O primeiro arcabouço usa super-resolução como um pré-processamento para a tarefa de segmentação semântica. O segundo arcabouço é uma abordagem unificada que treina as duas redes ao mesmo tempo enquanto compartilha suas funções de erro. Foram executados um conjunto extensivo de experimentos em dados de sensoriamento remoto com natureza e propriedades distintas. Para o conjunto de dados agriculturais de mapeamento de café, que contém apenas duas classes (café e não-café), o uso de imagens de baixa resolução alcançou apenas 50% de acurácia normalizada com taxa de aumento de 8 vezes. O arcabouço em dois estágios na mesma condição aumentou esse valor para 72%. O arcabouço unificado aumentou ainda mais esse valor para 77%, comparado aos 81% com dados de alta resolução. Para o conjunto de dados urbano de Vaihingen, usar super-resolução no arcabouço de dois estágios aumentou a acurácia de segmentação de carros de 19% para 58% com taxa de aumento de 8 vezes, enquanto o arcabouço unificado alcançou 65%. Nesse caso, com dados de alta resolução, a acurácia foi de 69%, o que não está distante dos resultados de super-resolução. Ambos os casos são exemplos de como super-resolução é capaz de recuperar detalhes de textura importantes (para plantações de café, por exemplo) e também é capaz de fazer ficarem mais claros objetos que eram difíceis de enxergar em uma representação de baixa resolução (como os carros). Os resultados mostram que super-resolução é efetiva para melhorar o desempenho de segmentação semântica em imagens aéreas de baixa resolução. Super-resolução não apenas é melhor que interpolação não-supervisionada, como também alcança resultados de segmentação semântica comparáveis a dados de alta resolução. Mesmo com pouco dado de treinamento, o uso dos arcabouços alcançou resultados melhores que usando interpolação bicúbica. Dessa forma, o uso de super-resolução se provou ser mais efetivo do que aplicar imagens de baixa resolução em uma rede neural de segmentação semântica. Isso é verdade especialmente para altos fatores de degradação, os quais são os casos em que super-resolução supera mais o desempenho de se usar diretamente dados de baixa resolução.

Palavras-chave: Sensoriamento Remoto, Super-Resolução, Segmentação Semântica.

Abstract

High-resolution aerial images are desirable for most of the deep-based remote sensing applications. This type of data, however, is not always accessible or affordable. On the other hand, coarse resolution remote sensing images, such as LANDSAT and MODIS, are easily found in public open repositories and, therefore, are widely used in many studies. The problem is that the amount of spacial information compressed into one single pixel in a low-resolution representation can compromise pattern recognition algorithms. Thus, the use of coarse-resolution data for automatic creation of thematic maps is very restricted since most of the deep-based semantic segmentation (a.k.a dense labeling) approaches are only suitable for subdecimeter data. Super-resolution is a classic computer vision problem that aims to restore the quality of degraded, low-resolution images. In this work, we design two frameworks in order to evaluate the effectiveness of deep-based super-resolution in the semantic segmentation of low-resolution remote sensing images. Our objective is to evaluate how effective is deep-based super-resolution to different levels of degradation, how it compares to unsupervised bicubic interpolation and if it is able to reconstruct small objects and, consequently, contribute to semantic segmentation improvement. The first framework uses super-resolution as a pre-processing step for the semantic segmentation task (two-stage framework). The second framework is an end-to-end approach that trains both networks at the same time while sharing their losses. We carried out an extensive set of experiments on remote sensing datasets with distinct nature and properties. For the agricultural dataset of coffee mapping, which only contains two labels (coffee and non-coffee), the use of low-resolution images achieved only 50% normalized accuracy with 8× up-scaling factor. The two stage framework with super-resolution in the same condition increased this value to 72%. The end-to-end framework further increased the value to 77%, compared to 81% of high-resolution data. For the urban dataset of Vaihingen, using super-resolution in the two stage framework increased the accuracy of car segmentation from 19% to 58% with 8× up-scaling factor, while the end-to-end framework achieved 65%. In this case, with high-resolution data, the accuracy was

69%, which is not far from the super-resolution result. Both cases are examples of how super-resolution is able to recover important texture details (for coffee crops, for example) and is also able to make more discernible small objects that were difficult to see in a low-resolution representation (such as cars). The results show that super-resolution is effective to improve semantic segmentation performance on low-resolution aerial imagery. It not only outperforms unsupervised interpolation but also achieves semantic segmentation results comparable to high-resolution data. Even with a few training data, the use of the frameworks still achieved better results than bicubic interpolation. Thus, using super-resolution has proven to be a more effective approach than directly inputting low-resolution images to a semantic segmentation network. This is especially true for high degrading factors, which are the cases that super-resolution surpasses more the performance of low-resolution data.

Palavras-chave: Remote Sensing, Super Resolution, Semantic Segmentation.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Remote sensing can be defined as the measurement at a distance of object properties on the Earth's surface. Such measurement can be done, for example, with the use of satellites, which provide a repetitive and consistent view of the Earth that is important to monitor short-term and long-term changes and the impact of human activities [Schowengerdt, 2006]. Many of these satellites are able to gather images directly in digital format, thus it can be processed by computers either for machine-assisted information extraction or for enhancement of its visual qualities [Richards and Jia, 1999]. There are operational satellite systems that sample nearly all available parts of the electromagnetic spectrum with dozens of spectral bands and with pixel sizes ranging from less than $1m$ to $1000m$ [Schowengerdt, 2006].

High-resolution aerial images are essential for many remote sensing applications, as they provide a finer representation of spatial boundaries [Pouliot et al., 2018a], more precise textures and can even display small objects that are barely visible in a low-resolution representation. Apart from high-end satellites, nowadays, there is another main way of directly acquiring high-resolution aerial images, which are drones. In reality, however, high-resolution image data is not always employable or accessible. Thus, relying on the two presented solution is often impracticable.

In order to map a large area, for instance, drones lack enough autonomy. On one hand, if using only one (or a few of them), the time required to map the whole area would be high. On the other hand, using many of them would increase the cost and the number of people involved in the process. This type of situation can make this option unviable.

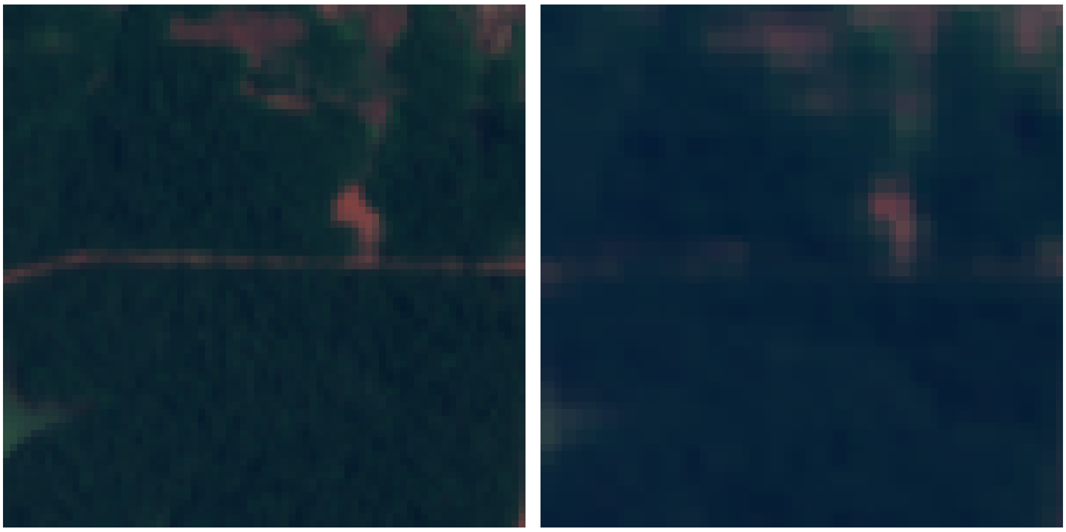
High-resolution satellites provide a more autonomous way, therefore capable of overcoming the problems presented by drones. However, their images are expensive and often present low temporal resolution, which can also make this option unviable.

With all these adversities in mind, due to data unavailability or high-cost reasons, the use of low-resolution images is often adopted in replacement of the high-resolution ones. Thus, an alternative for remote sensing applications is to get their data from low-resolution satellite imagery, which is cheap (usually free) and present a long history of acquisition (high temporal resolution). Landsat[1], for example, has a long record of images since 1984.

A main problem arises from the use of low-resolution images: the amount of important information compressed into one single pixel can compromise machine learning algorithms to detect or segment objects. As observed by Dai et al. [2016], semantic segmentation is one of the computer vision applications that is more severely affected by the input of low-resolution images. This application aims to predict a class for each pixel of the image, thus, if the objects are way too small (cars, for example) or have similar textures, such as different types of vegetation, the low-resolution will eventually cause cases of mislabeling, dropping the accuracy of the algorithm. Figures 1.1 and 1.2 illustrate examples of this situation. In Figure 1.1, we can see the difference between the same rural road across two different satellites (Sentinel-2 and Landsat 8) with different resolutions. When the spatial resolution is $10m$ per pixel (Sentinel-2) the road is easy to see, however as we decrease the spatial resolution, it becomes more difficult to detect. For the Landsat 8 satellite, which presents a spatial resolution of $30m$ per pixel, the road has almost disappeared. In Figure 1.2, we can see that there are three cars (yellow class), but only one of them was correctly labeled by a semantic segmentation network. Also, it is possible to see that big parts of the buildings (dark blue class) were mislabeled for impervious surfaces (white class). If the output were of a high-resolution, the cars would likely be correctly segmented and the buildings would not be easily mislabeled.

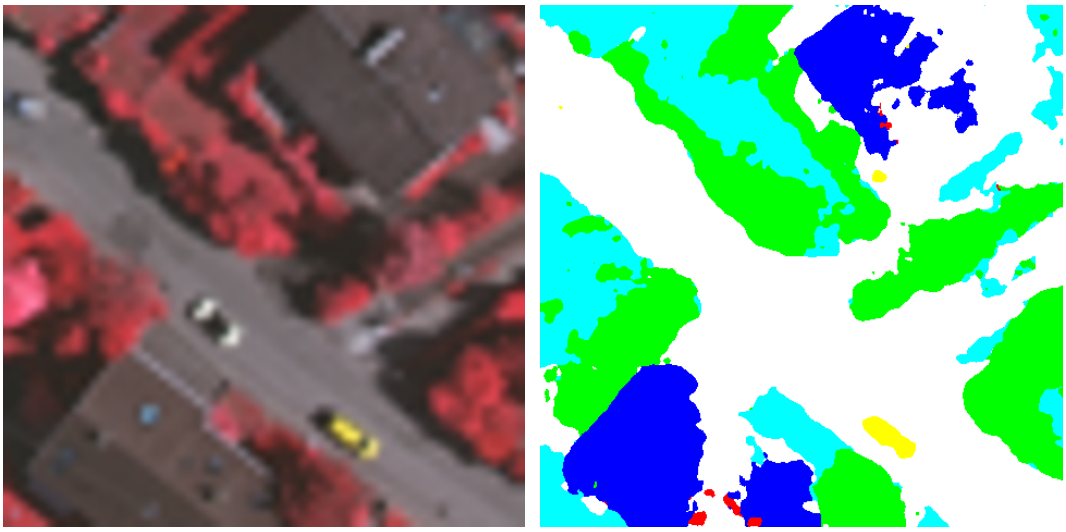
Remote sensing images are also highly subject to many adverse factors that conventional images usually do not suffer with, such as ultra-distanced imaging, atmospheric disturbance, and relative motion [Jiang et al., 2018]. Moreover, many satellites record different spectral bands with varying spatial resolutions, either by storage or transmission rate limit reasons [Lanaras et al., 2018]. Since it is highly desirable to have every spectral band of the sensors presenting the best achievable quality and at a high spatial resolution, a natural question to arise is: how can we effectively reconstruct low-resolution remote sensing imagery in order to improve different computer vision applications, such as semantic segmentation? In this work, we provide ways of achieving this goal with the use of super-resolution techniques.

[1] https://www.usgs.gov/land-resources/nli/landsat



(a) Sentinel-2 satellite (10m scale) (b) Landsat 8 satellite (30m scale)

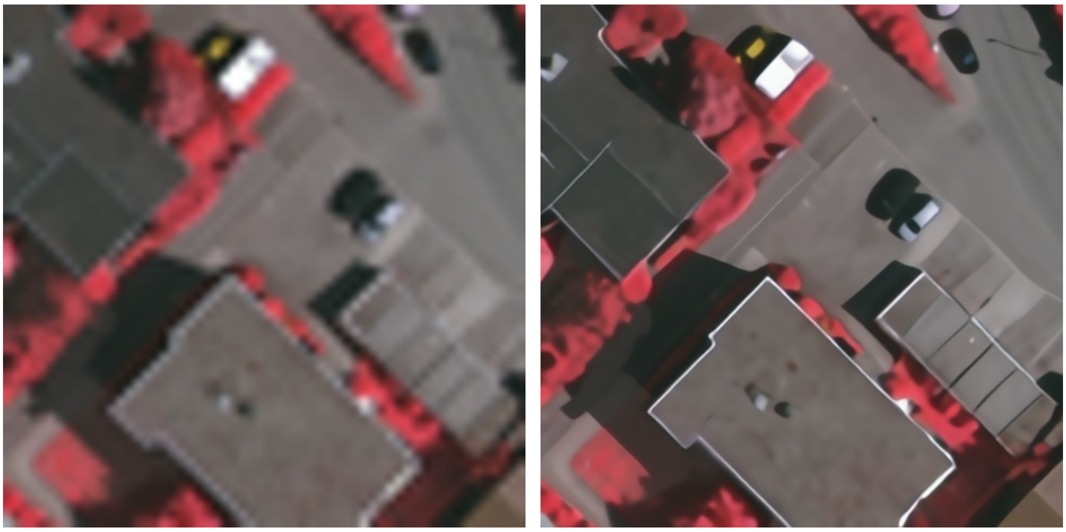
Figure 1.1: Example of lost of information for a rural road due to low resolution effects.



(a) Low-resolution image. (b) Semantic Segmentation Map.

Figure 1.2: Thematic map (b) generated from a low-resolution (a) input image with a SegNet [Badrinarayanan et al., 2017]. The image was up-sampled from its original 60 × 60 dimension to eight times more (480 × 480) with bicubic interpolation.

Super-resolution aims to construct a high-resolution image from a low-resolution input. Single image super-resolution (SISR) techniques are capable of achieving this objective by using only one single image as input, i.e., they do not require different versions of the same image in order to extract sub-pixel information. This is especially useful since most of the time we are interested in recovering the quality of unique images. Figure 1.3 illustrates an example of the reconstruction process performed by a super-resolution algorithm. Recently, convolutional neural networks (CNNs) have been successfully used in the super-resolution task with large improvements in image restoration, especially when compared to previous non-deep-learning methods. One of the biggest advantages brought by CNNs is that these methods do not require hand-crafted features that were typically necessary in previous algorithms.



(a) Low resolution input with original size of 60×60, up-sampled to 480×480 with bicubic interpolation.

(b) Reconstructed high resolution output of size 480×480 with 8× super-resolution scaling factor.

Figure 1.3: Example of reconstructed low resolution image.

Single-image super-resolution is used as a major tool for enhancing and restoring the quality of degraded images and its importance has increased with the growing demand for high-quality multimedia content [Pandey and Ghanekar, 2018]. It can be used, for example, to recover the original quality of images and videos on the Internet that normally lose information due to compression during transmission. High-resolution images are also important in the medical environment since they allow doctors to make a more accurate diagnosis. Considering remote sensing scenarios, as mentioned before, they are important due to the unavailability of high-quality data. Also, low-resolution images may be acquired due to hardware limitations and the requirement of high data rate transmission during satellite communication [Pandey and Ghanekar, 2018].

1.1 Objectives and contributions

The main objective of this work is to verify the performance of two frameworks that unite super-resolution and semantic segmentation methods to generate high-quality thematic maps for low-resolution remote sensing images. Specific objectives and contributions include:

- Evaluation of a two stage framework that uses super-resolution as a pre-processing step for a semantic segmentation task. This framework trains both networks separately.
- Evaluation of an end-to-end framework that trains both super-resolution and semantic segmentation networks at the same time.
- Verification of the performance of the aforementioned frameworks on remote sensing images when applied to different domains and datasets, such as urban and agricultural scenes.
- Evaluation of the effectiveness of super-resolution for the improvement of semantic segmentation on different levels of degradation.
- Comparison of the performance of a semantic segmentation network on low-resolution and reconstructed data.
- Evaluation of the robustness of super-resolution for small object detection and texture reconstruction.

We also review the main CNN-based methods and classify them according to their characteristics, presenting a brief discussion about the influences among them.

1.2 Outline

The remainder of this document is structured as follows. Chapter 2 presents a review of the most important super-resolution and semantic segmentation methods for both normal and remote sensing images. We also review recent similar works that evaluated super-resolution for the improvement of different pattern-recognition tasks. Chapter 3 introduces the frameworks mentioned in Section 1.1 along with their technical aspects. Chapter 4 presents the experiments that were conducted and the datasets we applied them on. We also show and discuss the obtained results. Finally, in Chapter 5 we conclude our work and discuss future possibilities.

Chapter 2

Literature Review and Related Work

Over the years, many methods have been proposed for both super-resolution and semantic segmentation. Recently, due to the advent of deep learning, the results of the methods for both problems were highly improved. This chapter presents a literature review on the most important recent works on super-resolution and semantic segmentation. We also discuss the works that combined super-resolution with another computer vision task in order to improve its results.

This chapter is organized into three parts. The first and second ones are focused on the literature review of the most relevant works up to date for the super-resolution and semantic segmentation tasks. Finally, in the third part, we present the works that also analyzed the use of super-resolution for the improvement of different pattern recognition tasks.

2.1 A Review on Super Resolution

Super-resolution is a classical problem in computer vision. The objective of this task is to generate a high-resolution image from a degraded, low-resolution one. This is an ill-posed problem since multiple possible high-resolution images can be mapped from a low-resolution input.

We divide the proposed methods in the literature into three categories:

1. **Shallow methods**: techniques that are not based on deep learning.

2. **Deep learning methods for RGB images**, i.e., common day-to-day images taken from conventional cameras. These methods may have been created with

convolutional neural networks (CNN) or generative adversarial networks (GAN).

3. **Deep learning methods for remote sensing images**: these methods are made mainly for aerial or satellite imagery. Some of them are capable of working with multispectral information.

Many shallow super-resolution methods have been proposed in the computer vision community. Early methods use very fast interpolations based on sampling theory but yield poor results, since those methods exhibit limitations in predicting detailed and realistic textures [Kim et al., 2016b; Lim et al., 2017]. These methods also usually suffer a dramatic drop in restoration performance with larger up-scaling factors [Hui et al., 2018].

Some of the most powerful shallow methods utilize statistical image priors or internal patch recurrence to avoid using external databases [Kim et al., 2016b; Lim et al., 2017]. However, this type of method usually only works well in images containing repetitive patterns and textures [Hui et al., 2018]. The external-example based methods learn a mapping between low and high-resolution patches from external datasets and usually focus on how to learn a compact dictionary or manifold space. While these approaches are effective, the extracted features and mapping functions are not adaptive, which may not be optimal for generating high-quality super-resolved images [Hui et al., 2018].

Since shallow methods were easily surpassed even by the first (and simplest) deep learning method, SRCNN [Dong et al., 2016], they are not currently able to achieve state-of-the-art performance and, therefore, will not be focused in this work.

Deep learning methods are capable of dealing with large amounts of data without requiring much domain-specific knowledge necessary to extract the relevant features for a problem's solution. Instead, those methods are capable of learning these features automatically given enough training data [LeCun et al., 2015]. The super-resolution task has the advantage of not requiring annotated datasets or any other type of manually annotated label. In fact, given a set of high-resolution images, one can already have enough information for training. That is because high-resolution images can be considered the label, while downgraded versions of them are the input for the network [Dong et al., 2016]. As the process of downgrading an image is usually easier than the opposite task, deep-based super-resolution networks have at their disposal an almost infinite amount of data. For training, in a general manner, many studies assume that the low-resolution image is a bicubic down-sampled version of its respective high-resolution version. However, other degrading factors such as blur, decimation, or noise can also be considered for practical applications [Lim et al., 2017]. This is an important remark, since manually downsampled low-resolution data may not accurately represent low-resolution images acquired from a real sensor.

We separated the many different deep learning based methods for SISR in three main categories: CNN-based for RGB images, GAN-based for RGB images and methods for remote sensing. The next subsections will present the state-of-the-art approaches for each of the categories.

2.1.1 CNN-based Methods for RGB Images

Convolutional Neural Networks (CNN) are the dominant approach for most of the problems involving images [LeCun et al., 2015]. Naturally, it would be expected that these types of network would tackle the super-resolution problem. Given that CNNs are capable of extracting important features from the training data, they are also able to use such features in order to reconstruct high-quality pixel information in degraded images [Dong et al., 2016].

An overview of the most import CNN-based methods for super-resolution is presented in Figure 2.1. The arrows represent a transitive relation and point to methods that were somehow influenced by the ones that the arrows come from. These influences will be later discussed, but they mainly represent techniques for CNNs applied to super-resolution. For example, SRCNN [Dong et al., 2016] directly influenced VDSR [Kim et al., 2016a] and indirectly influenced LapSRN [Lai et al., 2017], EDSR [Lim et al., 2017], etc. Similarly, LapSRN [Lai et al., 2017] was directly influenced by ESPCN [Shi et al., 2016] and VDSR [Kim et al., 2016a], while being indirectly influenced by SRCNN [Dong et al., 2016]. Also, each column groups methods with quantitatively similar results. This was done by comparing the peak signal-to-noise ratios (PSNR) results for the usual super-resolution benchmark datasets, such as *Set14* [Zeyde et al., 2012] and *Urban100* [Huang et al., 2015]. The grouping means that, for example, VDSR [Kim et al., 2016a] (and those in its column) can be considered more effective than SRCNN [Dong et al., 2016]. As another example, WDSR [Yu et al., 2018] can be considered more effective than those on previous columns (such as EDSR [Lim et al., 2017], IDN [Hui et al., 2018], DRCN [Kim et al., 2016b] and so on), less effective than the methods on the following columns (such as RCAN [Zhang et al., 2018a] and FRN [Kwak and Son, 2019]) and equivalent in terms of results to those in the same column (such as D-DBPN [Haris et al., 2018a] and RDN [Zhang et al., 2018b]). The methods marked as yellow are the ones developed for remote sensing data and will be discussed in a later subsection.

CNN-based methods: Dong et al. [2016] proposed the first deep learning based

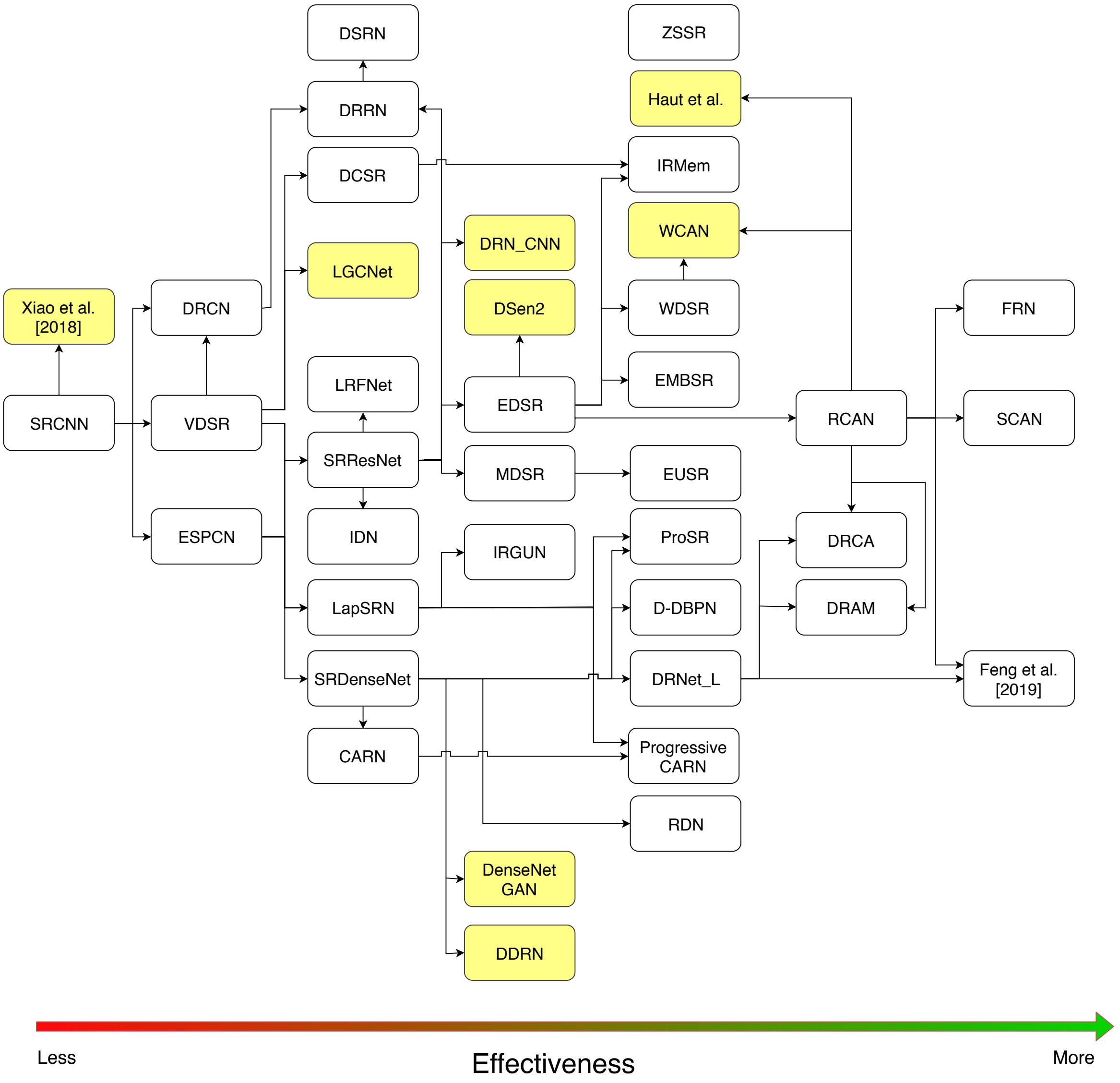


Figure 2.1: An overview of the most relevant recent CNN-based methods for super-resolution.

method for SISR, named SRCNN. Being the first CNN approach for super-resolution, their work is an influence on all the following methods. Dong et al. [2016] verified that the sparse-coding-based method [Yang et al., 2010] pipeline could be modeled in a CNN framework, which is: (*i*) patch extraction, (*ii*) low-resolution dictionary encoding, (*iii*) high-resolution dictionary reconstruction and *(iv)* image reconstruction. Their CNN only contains three convolutional layers, one representing each step of the aforementioned pipeline. Dong et al. [2016] attempted to prepare deeper models but failed to observe superior performance. In some cases, deeper models even gave an inferior performance. For this reason, they concluded that deeper networks do not result in better performance. However, as demonstrated in the papers that followed

SRCNN [Kim et al., 2016a,b; Lim et al., 2017; Haris et al., 2018a; Yu et al., 2018; Wang et al., 2018b; Kim and Lee, 2018; Seif and Androutsos, 2018a; Park et al., 2018], it is possible to achieve better results with deeper models. Therefore, although reaching state-of-the-art performance for its time, SRCNN has some flaws that prevent it from reconstructing even better high-resolution images. Among these flaws, the most notable are:

- The input for the network is a bicubic-up-sampled low-resolution image: this not only increases the computational cost (since the input image has the same size of the output) but also inserts bad artifacts to the input (caused by the interpolation) that can negatively affect the output. Instead, the network could work on the original low-resolution image size and learn to reconstruct it with deconvolutional or sub-pixel up-sampling layers, for example [Lim et al., 2017].

- There is no residual learning: residual learning techniques (by adding skip connections to the network) were shown later to be essential for super-resolution methods [Kim et al., 2016a,b; Lim et al., 2017; Haris et al., 2018a; Yu et al., 2018; Wang et al., 2018b; Kim and Lee, 2018; Seif and Androutsos, 2018a; Park et al., 2018], since they allow more layers to be used without harming the training procedure (by avoiding gradient problems). They also make the network learn only what is necessary to recover high-frequency details from the original low-resolution image, instead of learning to construct a whole new high-resolution image [Kim et al., 2016a].

Residual learning methods: Inspired by SRCNN [Dong et al., 2016] and VGG-net used for ImageNet classification [Simonyan and Zisserman, 2015], Kim et al. [2016a] proposed the method named VDSR. This method was the first to use residual learning, which causes the network to learn only the difference between high and low-resolution images, thus improving both convergence time and reconstruction results. This motivates the usage of deeper networks. The methods in Figure 2.1 influenced by VDSR are all characterized by the use of residual learning. VDSR is able to perform multi-scale super-resolution, which allows arbitrary scaling factors (including fractions), while SRCNN would need a different training procedure for each scaling factor. In order to allow the network to operate in a multi-scale, the training dataset is composed of several specified scales, which can be done by applying bicubic interpolation with different scales on the same set of high-resolution images. Much deeper than SRCNN, which uses 3 layers, VDSR can efficiently solve the SISR problem with 20 layers.

Recursive methods: DRCN [Kim et al., 2016b] is another method proposed by the same authors of VDSR. It repeatedly applies the same convolutional layer many times in a recursive manner, which re-utilizes the same parameters. In order to alleviate the difficulty of training, all recursions are supervised. This means that feature maps after each recursion are used to reconstruct the target high-resolution image. For better training, all predictions resulting from different levels of recursions are combined to deliver a more accurate final prediction [Kim et al., 2016b]. Skip connections are applied in order to perform residual learning like in VDSR. The methods that were inspired by DRCN are all characterized by the use of recursive layers. Namely, those methods are DRRN [Tai et al., 2017] and DSRN [Han et al., 2018].

Late up-sample methods: Both VDSR and DRCN use as input bicubic up-sampled low-resolution images that have the same size as the output. Thus, just like SRCNN, the amount of computation necessary is increased and bad artifacts can be added to the image because of the interpolation. Differently from this aspect, ESPCN [Shi et al., 2016] proposed inputting the original low-resolution image to the super-resolution network, instead of first upsampling it with a simple method. Thus, in this method, the feature maps are extracted in the low-resolution space. The last layer of this network is a sub-pixel convolution that learns how to up-scale the image by merging the last low-resolution feature maps. This approach reduces the computational and memory complexity [Shi et al., 2016]. In Figure 2.1, all methods influenced by ESPCN operate in the low-resolution space, i.e., they do not input a bicubic up-sampled image.

Dilated convolution methods: DCSR [Zhang et al., 2019] introduced the use of dilated convolution for the super-resolution task. In Figure 2.1, all methods influenced by DCSR also present this characteristic (IRMem [Chen et al., 2018b]). Dilated convolutions allow the expansion of receptive fields without increasing the number of parameters or computational complexity. More specifically, in DCSR the authors mix both dilated and standard convolutions in the layers, concatenating the extracted feature maps. This approach removes blind spots in the receptive field [Zhang et al., 2019].

Resnet-based methods: Resnet [He et al., 2016] was a successful network for the image recognition task and was adapted to the super-resolution problem by Ledig et al. [2017] with the method named SRResNet. Unlike VDSR [Kim et al., 2016a], which uses a single residual mapping from the input to the output of the network, methods like SRResNet apply this technique along all of the network structure. This architecture helps with the training of the network since it is easier to optimize. This happens because the skip connections make the network more resistant to the vanishing gradient problem. The methods influenced by SRResNet also share its

Resnet-based architecture, such as LRFNet [Seif and Androutsos, 2018b] (with large receptive fields and 1D kernels) and IDN [Hui et al., 2018] (with information distillation blocks). IRMem [Chen et al., 2018b] is a method that applies dilated convolutions, like DCSR [Zhang et al., 2019], and presents a resnet-based architecture, with the difference of grouping feature maps in a gate at the end of the residual block. DRRN [Tai et al., 2017] is another special case which merges the idea of both resnet-based architecture and recursive layers.

Two of the most important methods based on resnet were proposed by Lim et al. [2017]. They optimized the residual modules in existing conventional networks and proposed the methods EDSR and MDSR, which ranked, respectively, first and second on the NTIRE2017 Super-Resolution Challenge [Timofte et al., 2017]. One of the modifications proposed by Lim et al. [2017] was the removal of batch normalization layers from the network. As they mention, the normalization of features gets rid of range flexibility and their experiments demonstrated that this simple change substantially improved the network's performance. This choice became a pattern on most of its following works [Wang et al., 2018b; Haris et al., 2018a; Yu et al., 2018]. EDSR is the single-scale model proposed by Lim et al. [2017]. They also apply sub-pixel up-sampling layers as in ESPCN Shi et al. [2016]. EDSR was an especially important method for the super-resolution literature due to its influence on later works. EMBSR [Park et al., 2018], for example, build their method with a few alterations in the EDSR model.

Based on the idea that pre-training high scaling networks on a smaller scale one is beneficial, which demonstrates that super-resolution at multiple scales is an inter-related task, Lim et al. [2017] also proposed the MDSR model. Although having more parameters than EDSR ($1.5M$ against $3.2M$), MDSR can super resolve the low-resolution input in three different scales, while the single-scale model would need three training procedures in order to do so. Like EDSR, MDSR was also used as a base for some later approaches, such as EUSR [Kim and Lee, 2018].

Progressive up-sampling networks: LapSRN [Lai et al., 2017] applied in its architecture a similar approach do ESPCN [Shi et al., 2016]. But instead of using one single layer to up-sample the feature maps to the desired resolution, LapSRN progressively up-samples to intermediate sizes the feature maps in the middle layers. This approach allows multi-scale predictions in one forward step. According to Lai et al. [2017], using one single up-sampling step increases the difficulties of training for large scaling factors. Another difference from ESPCN is that LapSRN uses transposed convolution layers in instead of pixel-shuffling layers as the up-sampling technique. All the methods influenced by LapSRN in Figure 2.1 apply some kind of progressive up-sampling structure, such as IRGUN [Sharma et al., 2018].

Densenet-based methods: Similarly to how SRResNet [Ledig et al., 2017] adapted the well-known Resnet [He et al., 2016] to the super-resolution task, SR-DenseNet [Tong et al., 2017] and the methods influenced by it (as seen in Figure 2.1) adapted Densenet [Huang et al., 2017] for super-resolution. This type of architecture uses as inputs the feature-maps of all preceding layers are for each layer. Like Resnet, this technique also makes the network robust to the vanishing gradient problem, but it also encourages feature reuse even more [Huang et al., 2017]. For super-resolution, this is an effective way of combining low and high-level features to improve the reconstruction performance [Tong et al., 2017]. Densenet-based architectures recently became a common pattern for many super-resolution algorithms. From Figure 2.1 we can mention CARN [Ahn et al., 2018a], RDN [Zhang et al., 2018b] and DRNet_L [Wen et al., 2018]. There are also effective methods that united both the densenet architecture and the progressive up-sampling technique from LapSRN [Lai et al., 2017], such as Progressive CARN [Ahn et al., 2018b] and ProSR [Wang et al., 2018b].

Back projection methods: The super-resolution competition NTIRE2018 brought a large scaling factor (8×) challenge for classic bicubic-down-sampled images. In this category, Haris et al. [2018a] were the winners with their method named D-DBPN, which constructs both up and down-sampling stages. D-DBPN focuses on projecting the high-resolution features back to the low-resolution spaces using down-sampling layers. They also apply the densnet strategy mentioned before by densely connecting each up and down-sampling stage, which encourages feature reuse. This enables the network to reconstruct the high-resolution image using a deep concatenation of the feature maps from the previous steps [Haris et al., 2018a]. Both up and down projection layers concatenate the feature maps of all preceding units (up-sampling blocks take from down-sampling blocks and vice-versa) as input, and their own feature maps are used as inputs into all subsequent units. Inside each of these blocks, a series of convolutions and deconvolutions with skip connections are used in order to generate the feature maps.

Wide activation methods: WDSR [Yu et al., 2018] is another method that achieved impressive results on the NTIRE2018 competition. While D-DBPN ranked first place on the classic 8× classic bicubic track, WDSR ranked first in all other three tracks, which up-scaling factor is 4× and include different types of degradation on the images, thus making the super-resolution task even more difficult. In their work, Yu et al. [2018] affirm that ReLUs hinder information flow from shallow layers to deeper ones. Thus, they propose to expand features before the activation functions in a way that does not increase the model's complexity or the number of parameters. In order to do this, WDSR slims the features of residual identity mapping while expands the features before activation. Along with the aforementioned process of wide activation, Yu et al. [2018] verified that weight normalization, which is a reparameterization of the weight vectors that decouples the length of those vectors from their direction, can be useful to achieve better accuracy without introducing dependencies between examples in a mini-batch.

Zero-shot methods: Shocher et al. [2018] affirm that externally supervised methods perform well on data satisfying the conditions they were trained on, while their performance decreases when these conditions are not satisfied. The reason for this is that training images are usually generated with a bicubic interpolation, without considering artifacts usually present on real-world images (like sensor noise and compression artifacts). Because of that, Shocher et al. [2018] proposed a zero-shot method, named ZSSR, which exploits the internal recurrence of information within a single image. Thus, it does not rely on prior image examples or prior training. It trains a small image-specific CNN at test time on examples extracted from the low-resolution input image itself [Shocher et al., 2018]. In ZSSR, the low resolution is furthermore degraded into an image with an even smaller resolution, X. The training is then performed between the pair X and the low-resolution, and the final network is applied to the low-resolution image in order to construct the high-resolution version. Since only one image is used, Shocher et al. [2018] employ techniques of data augmentation to create different versions of the same image.

Channel-attention networks: In channel-attention, trainable weights are multiplied to each channel of the image Jang and Park [2019]. This type of technique was first applied to super-resolution by Zhang et al. [2018a]. Based on the idea that low-frequency information is treated equally across channels in most of the super-resolution approaches, which hinders the representational power of the networks, Zhang et al. [2018a] proposed the method called RCAN. In this method, the channel-attention mechanism adaptively learns channel-wise features while considering interdependencies among channels [Zhang et al., 2018a]. They also employ a residual-in-residual architecture that is similar to resnet-based approaches by stacking residual architectures inside residual architectures. Recently, the use of channel-attention modules has been important to state-of-the-art methods for super-resolution. Jang and Park [2019] and Wang et al. [2019a] further explored such technique by also employing densenet-based modules/architecture. Xu and Li [2019] is another state-of-the-art method highly influenced by RCAN [Zhang et al., 2018a]. Their architecture considers spatial and color features as complementary domains, which allows the network to focus on recovering high-frequency components and sharp color information in the generated image [Xu and Li, 2019].

Kwak and Son [2019] also employ the channel-attention and residual-in-residual techniques in their method named FRN. They employ a down-sampling module at the beginning of the network in order to exploit GPUs more efficiently (because of size-reduced features) with an up-sampling module at the end. Unlike D-DBPN [Haris et al., 2018a], the down and up-sampling operation are not performed along with the whole network. Finally, Feng et al. [2019] propose a network with a U-Net structure with cascading blocks containing channel-attention mechanisms. These blocks are connected with a densenet-based architecture.

Considering an overview of Figure 2.1, it is possible to say that the first two columns contain the basic methods that were not yet close to exploring the full potential of super-resolution. The third column is where the main architectures for the following methods were first presented and shown to be effective, such as resnet-based and densenet-based. The fourth column contains the last methods that were limited to less than 4 times super-resolution or bicubic downsampled data. In the fifth column, the methods started to become able to perform super-resolution on large scaling factors (up to 8 times) or to deal with more realistic downsampling kernels. Finally, the last two columns contain the current state-of-the-art methods that improved even more the characteristics of their predecessors.

2.1.2 GAN-based Methods for RGB Images

Generative Adversarial Networks (GAN) are a framework in which one simultaneously trains two models: a generative model G that captures the data distribution and a discriminative model D that estimates the probability that a sample came from the training data rather than G. The training procedure for G is to maximize the probability of D making a mistake [Goodfellow et al., 2014].

GANs provide a powerful framework for generating plausible-looking natural images with high perceptual quality [Ledig et al., 2017]. These are characteristics that are highly relevant to super-resolution problems. Thus, some works also explored the SISR problem using this powerful framework.

The aforementioned methods usually focus on minimizing the mean squared error or mean absolute error. Ledig et al. [2017] showed that this results in estimates that have high peak signal-to-noise ratios (PSNR), but often lack high-frequency details that make images more perceptually satisfying. This happens because minimizing these loss functions encourages generating overly-smooth textures [Ledig et al., 2017]. For this reason, they presented a generative adversarial network (SRGAN) capable of inferring photo-realistic images for 4× upscaling factors.

Ledig et al. [2017] proposed a perceptual loss function which consists of an adversarial loss and a content loss. The former uses a discriminator network that is trained to differentiate between the reconstructed and original images. They also used a content loss motivated by perceptual similarity instead of similarity in pixel space. With a mean-opinion-score (MOS) test, they proved that their approach is able to generate images with a better perceptual quality even when the PSNR values are lower than other methods.

After SRGAN, some papers also explored the GAN framework in order to create visually appealing high-resolution images. Among these works, Yuan et al. [2018] proposed an unsupervised Cycle-in-Cycle network structure, named CinCGAN. First, their method maps the noisy and blurry input a noise-free low-resolution space. The result is then up-sampled with a pre-trained deep model (such as EDSR) and, finally, they fine-tune the two modules in an end-to-end manner to get the high-resolution output [Yuan et al., 2018]. Therefore, their pipeline consists of learning a mapping from a low-resolution image to a noise-free low-resolution image, which has a known down-sample kernel: a bicubic interpolation. Then an existing SR model is used to super-resolve the clean image to the desired resolution. The fine-tune procedure is performed on both models at the end. As an unsupervised training procedure is performed on CinCGAN, the problem of complex degradation functions of real-world images is surpassed, like in ZSSR [Shocher et al., 2018]. Also, it removes the necessity of high-resolution images for training, which are not available in many practical scenarios.

Some other GAN based approaches for SISR include Wang et al. [2018a] and Wang et al. [2018b]. The first one uses categorical priors to characterize the semantic class of a region to constrain the super-resolution solution space. This is done with a Spatial Feature Transform approach that alters the behavior the network by transforming the features of some intermediate layers conditioned on semantic segmentation probability maps. The second method, ProGanSR [Wang et al., 2018b], is the GAN version of ProSR (both proposed in the same paper) that also employs a progressive architecture and curriculum learning.

2.1.3 Discussion on Super-Resolution for Remote Sensing Images

Some works in the literature have addressed the super-resolution problem for remote sensing images. These methods basically follow the same procedure as those for standard images. However, remote sensing images are able to capture information from more than just the visible spectrum. For this reason, this type of image often requires more adequate methods.

Lanaras et al. [2018] proposed two methods to super-resolve the lower-resolution ($20m$ and $60m$ ground sampling distance - GSD) bands of the Sentinel-2 satellite to $10m$ GSD, named DSen2 and VDSen2. Their methods are not overfitted to a particular context, which means the training takes samples from different places around the world. Their main assumption is that up-sampling from $20m$ to $10m$ GSD can be learned from ground truth images at $40m$ and $20m$ GSD, and the same is valid for the $60m$ to $10m$ case. In other words, they presume that the spectral correlation of the image texture is scale-invariant over a limited range of factors [Lanaras et al., 2018].

Like in classic images, data for training can be generated by down-sampling original images and using them as input for the network, while the ground truth image is used as the desired output. For this task, they down-sample the original data by first blurring it with a Gaussian filter, followed by averaging windows of pixels of different sizes for each GSD. This causes the generation of two datasets: the first consists of high-resolution images at $20m$ GSD and the down-sampled low resolution images of $40m$ GSD acquired from the $20m$ and $10m$ GSD bands (which are used to train a network for a factor of $\times 2$), and the second one contains images with $60m$, $120m$ and $360m$ GSD, down-sampled from the original $10m$, $20m$ and $60m$ data (which are used to learn a network for a factor of $\times 6$).

In order to transfer the high-frequency content to the low-resolution input bands, Lanaras et al. [2018] also input the high-resolution bands to the network. This means that $10m$ GSD bands are used with the $20m$ in the $\times 2$ network. Also, $10m$ and $20m$ bands are used with the $60m$ GSD bands in the $\times 6$ network. Their network structure is based on EDSR [Lim et al., 2017], with the difference that they work on bicubic up-sampled inputs in order to match the high ($10m$ GSD) resolution bands that also serve as input.

Jiang et al. [2018] affirm that incremental depth in a deep CNN framework causes loss of information, in special for remote sensing images, which have a complicated degradation process, low ground object resolution, and weak textures. Naming their method DDRN, Jiang et al. [2018] used a recursive strategy similar to Kim et al. [2016b] and ultra-dense-connections, similarly to Tong et al. [2017], which enhances the representational power of the network and releases the memory burden.

Lei et al. [2017] affirm that in typical CNN models, lower convolutional layers share small size of receptive fields and focus more on local details, while in deeper layers, bigger receptive fields are accumulated, which covers a larger area of data. To better compensate both local and global information, they created the method named LGCNet with a multifork structure that learns multiscale representations of remote sensing data, including both local details (such as edge and contours of an object) and global priors. Lei et al. [2017] tested their network on the UC Merced dataset, which contains RGB images with relatively high spatial resolution. The results proved that their approach surpasses earlier methods, such as SRCNN and VSDR, however, no comparison is made with more elaborate networks.

A few more works adapt already well-established methods for standard images in order to apply them to remote sensing data. Among these works, we can mention Pouliot et al. [2018b] (DRN_CNN), which adapts a resnet-based architecture in order to super-resolve Landsat images from the training with Sentinel-2 data. Xiao et al. [2018] applied an SRCNN-like network for Jilin-1 satellite data. Bosch et al. [2018] (DenseNet GAN) adapted the densenet-based architecture in a GAN framework. Gu et al. [2018] (WCAN) and Haut et al. [2019] took inspiration from the successful channel-attention approach, while WCAN also used the wide activation technique as in WDSR Yu et al. [2018].

2.2 Semantic Segmentation Methods

Semantic segmentation is another classic computer vision problem. The objective of this task is to predict for every pixel the class it belongs to. Semantic segmentation has many applications, such as scene understanding and autonomous driving [Badrinarayanan et al., 2017]. Like super-resolution, semantic segmentation also had the state-of-the-art changed by the introduction of deep learning, especially with the use of fully convolutional networks (FCNs). Long et al. [2015] adapted classification networks, such as AlexNet, VGG net, and GoogLeNet into FCNs and fine-tuned their learned representations to the segmentation task. This way, they achieved state-of-the-art results by a relative improvement of 20% mean intersection over union compared to previous works.

Ronneberger et al. [2015] proposed U-Net, a method that works with very few training images by relying on a strong use of data augmentation, which allows the network to learn invariance to elastic deformations. Their method consists of a contracting path and an expansive path, resembling an FCN, with the difference that they use skip connections to link low and high-level feature maps across resolutions. Like Long et al. [2015], Ronneberger et al. [2015] also achieved state-of-the-art performance on the selected datasets.

As the previous works mainly focused on adapting deep architectures designed for classification to pixel-wise labeling, Badrinarayanan et al. [2017] proposed a method

(Segnet) that was created to be more optimized for the semantic segmentation task, while also being efficient both in terms of memory and computational time. Unlike U-Net, for example, Segnet stores the max-pooling indices of the feature maps and uses them in its decoder network to achieve good performance, instead of storing the encoder network feature maps in full [Badrinarayanan et al., 2017]. With this approach, Segnet managed to surpass the results of the previous method without an increase in computational cost.

Chen et al. [2017] proposed DeepLabV3. This method uses atrous convolutions, which is an approach that allows the explicit adjusting of the filters' field-of-view. In order to segment objects at multiple scales, the authors designed modules that employ atrous convolutions in cascade or in parallel to capture multi-scale context. Expanding this method furthermore, Chen et al. [2018a] adds a decoder module to improve the segmentation results. This encoder-decoder approach for the method is named DeepLabV3+. The encoder module, which is the base DeepLabV3 [Chen et al., 2017] block, encodes the multi-scale contextual information with atrous convolutions at multiple scales, while the decoder module refines the segmentation results along object boundaries [Chen et al., 2018a]. Allowing different backbones to work in the method, such as Resnet [He et al., 2016] and Xception [Chollet, 2017], DeepLabV3+ can become faster and stronger, which allowed them to achieve state-of-the-art performance.

Semantic segmentation has also been applied to remote sensing data. Many methods base themselves on already consolidated networks and propose some modifications or adaptations. Sherrah [2016], for example, proposed methods based on fully convolutional networks [Long et al., 2015], while Audebert et al. [2017] and Marmanis et al. [2018] proposed methods based on Segnet [Badrinarayanan et al., 2017].

2.3 Super-Resolution for Image Analysis Improvement

Despite the growing interest in super-resolution and semantic segmentation, almost no study has yet been made evaluating the performance of methods for both problems together, especially for aerial imagery. Dai et al. [2016] evaluated super-resolution methods for several vision tasks, which were edge detection, semantic segmentation, digit recognition, and scene recognition. Their experiments showed that applying super-resolution to input images of other vision systems does improve their performance when the input images are of low-resolution. They also verified that standard perceptual criteria used to evaluate super-resolution methods (such as PSNR and SSIM) correlate quite well with the usefulness for the vision tasks, but that they may not be the most accurate proxies. For the semantic segmentation task, Dai et al. [2016] improved the average precision of pixels by 9.1% when using SRCNN [Dong et al., 2016] compared to inputs up-sampled by bicubic interpolation on the MSRC-21 dataset. Although having a similar purpose to us, it is important to remark that not only they did not make an evaluation on aerial imagery, but they also applied methods for both super-resolution and semantic segmentation that are no longer close to the state-of-the-art.

Haris et al. [2018b] proposed a more elaborated framework, named Task-Driven Super-Resolution. This framework unifies both super-resolution and object detection tasks in an end-to-end training, which incorporates a tradeoff between detection and reconstruction losses. The motivation for this is the difference between machine and human perception, thus the framework creates images that not necessarily correspond to the highest super-resolution evaluation criteria (usually PSNR), but that, in return, present features that help the machine algorithm to detect the objects. For this purpose, they used D-DBPN [Haris et al., 2018a] for super-resolution and SSD [Liu et al., 2016] for object detection. In terms of detection performance, their results surpassed the mean average precision (mAP) of bicubic interpolation more than three times for the simplest test (without blur or noise) and up to around eight times for the hardest test (with noise) on 8× up-scaling. Like in Dai et al. [2016], the tests were not conducted on aerial images.

In Ferdous et al. [2019] and Shermeyer and Van Etten [2018], the authors used super-resolution to assist object detection performance in aerial imagery. Both used SSD [Liu et al., 2016] for detection, but while the former applied SRGAN [Ledig et al., 2017] as the super-resolution method, the latter chose VDSR [Kim et al., 2016a]. Shermeyer and Van Etten [2018] verified that super-resolving native $30cm$ GSD imagery to $15cm$ yields a 16 to 20% improvement in detection mAP on the xView Dataset. They also noted that there is less gain at coarser resolutions and justified that this happens because the algorithm is unable to find enough unique discriminating features to adequately reconstruct a high-resolution image. Ferdous et al. [2019] applied their tests on the VEDAI dataset and achieved around 27% more mAP in 4× up-scaling compared to a low-resolution input.

Chapter 3

Methodology

In this chapter, we introduce the two frameworks we have proposed in this work. Both of them are composed of two main blocks: a super-resolution network and a semantic segmentation network. The first framework uses super-resolution as a pre-processing step for semantic segmentation, thus the two networks are trained separately. The second framework is an end-to-end approach that trains both networks at the same time, taking into consideration the loss of the two tasks. We start by explaining each individual network (for super-resolution and semantic segmentation) and then we present the proposed frameworks.

3.1 Base Methods

3.1.1 The Super-resolution Network

D-DBPN [Haris et al., 2018a] was the chosen super-resolution network to be employed in the frameworks. As mentioned in Chapter 2, this network is influenced by the densenet-based architecture and it was the first main method to employ up and down-sampling stages along all the network body. This method gained attention for being the winner of the first track of the 2nd NTIRE challenge on single image super-resolution. More specifically, this track evaluated super-resolution methods using the bicubic down-scaling with 8× degradation factor. Before D-DBPN, most of the methods were not capable of accurately performing super-resolution on such high scaling factors.

The main characteristic of the D-DBPN network is the error feedback mechanism. Haris et al. [2018a] affirm that when operating on large scaling factors, feed-forward architectures have access to a limited amount of features in the low-resolution space. For this reason, D-DBPN focuses on projecting the high-resolution features back to the low-resolution spaces using down-sampling layers. This allows the network to guide the image reconstruction by calculating the projection error from the up and down-sampling blocks. The different ways of projecting back to another low-resolution representation enrich the knowledge of the network, which learns different ways of up-sampling the features. This additional knowledge is made possible with the use of a deep concatenation of high-resolution features. By concatenating at the end of the network the feature maps generated by all the previous up-sampling modules, D-DBPN reconstructs a final high-resolution image with different low to high-resolution mappings. This is further explored with the use of a densenet-based architecture, which encourages feature reuse.

Figure 3.1 shows the full D-DBPN [Haris et al., 2018a] architecture. Figures 3.2 and 3.3 illustrate the up and down projection modules proposed by Haris et al. [2018a]. The first two layers of the network are used to extract basic features from the low-resolution input image. The body of the network is composed of several up and down projection modules densely connected in order to encourage feature reuse. The tail of the network reconstructs the final image with the use of all previous high-resolution feature maps generated.

The up-projection blocks (Figure 3.2) take low-resolution inputs and bring them to a high-resolution representation with the use of deconvolution. These high-resolution feature maps are projected back to the low-resolution representation with a convolution layer. The difference between the input and the newly generated low-resolution feature maps is taken in order to produce a different intermediate low-resolution information, which is brought back to the high-resolution space and summed to the result of the first deconvolution. The result of all this procedure will be then sent to both the next down-projection block and to the concatenation of high-resolution features at the end. The procedure for the down-projection blocks (as in Figure 3.3) works similarly to the up-projection, but performing the opposite operation.

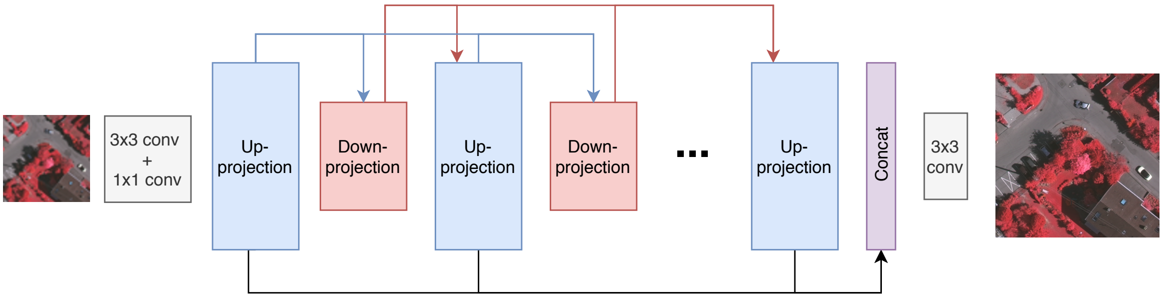


Figure 3.1: Illustration of D-DBPN [Haris et al., 2018a] architecture.

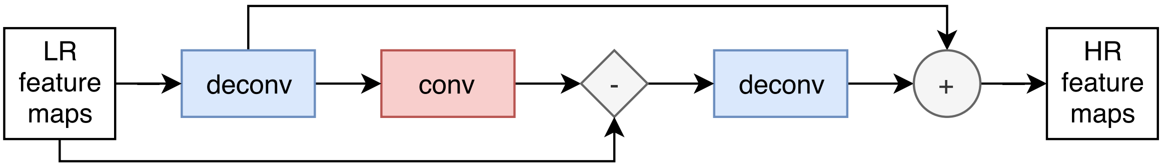


Figure 3.2: Illustration of D-DBPN's [Haris et al., 2018a] up projection block.

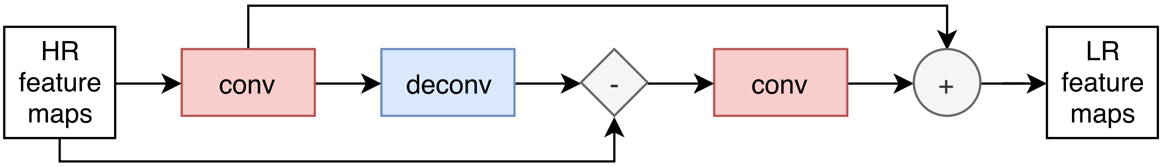


Figure 3.3: Illustration of D-DBPN's [Haris et al., 2018a] down projection block.

In order to train the super-resolution network, we need pairs of corresponding low and high-resolution images. Two sensors of different image quality or in different heights can be used to get those pairs. However, it is also possible to automatically generate low-resolution images by degrading the high-resolution ones.

We use the same default network configuration proposed in D-DBPN paper [Haris et al., 2018a]. Thus, for 4× enlargement we use 8×8 convolutional layer with four striding and two padding, while for 8× enlargement we use 12×12 convolutional layer with eight striding and two padding. As the final network in Haris et al. [2018a], we set the number of back-projection stages (up + down projection modules) to 7.

Most super-resolution methods aim to maximize the Peak signal-to-noise ratio (PSNR). PSNR is commonly used to measure the reconstruction quality of lossy transformation, such as image compression [Wang et al., 2019b]. For super-resolution, it serves as a metric to quantitatively evaluate image restoration quality [Dong et al., 2016]. The PSNR is calculated using the mean squared error (MSE) between the high-resolution image I_{HR} and the reconstructed image from its low-resolution version, I_{SR}, as in Equation 3.1.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(I_{HR}(i) - I_{SR}(i))^2, \tag{3.1}$$

where N is the total amount of pixels in the image. The PSNR is then calculated as in Equation 3.2.

$$PSNR = 10 \cdot log_{10}(\frac{255^2}{MSE}), \tag{3.2}$$

where 255 is the maximum pixel error considering an 8 bits integer representation.

It was pointed by Dong et al. [2016] that minimizing loss functions such as mean squared error (MSE or L2) favors high PSNR values. However, Lim et al. [2017] empirically found that the mean absolute error (L1) loss provides better convergence than L2. Thus, D-DBPN is trained with the mean absolute error loss (L1 loss), defined as in Equation 3.3.

$$L1 = \frac{1}{N}\sum_{i=1}^{N} |I_{HR}(i) - SR(I_{LR}(i); \theta_{sr})|, \tag{3.3}$$

where I_{HR} is the ground-truth high-resolution image, $SR(.)$ is the output of D-DBPN (SR function) given the low-resolution input image I_{LR} and all the parameters of the network, θ_{sr}.

The choice for D-DBPN comes from the fact that it was the state of the art method for high-scale restoration (up to 8 times). Begin able to recover better details for such a high factor is especially important for aerial images, since they can present ground sample distance (which can be defined as the distance a pixel represents in reality) varying from a few centimeters (drone images, for example) to many meters (such as in the LANDSAT-8 satellite). It was also successfully used in Haris et al. [2018b] in their task-driven framework.

3.1.2 The semantic segmentation network

The semantic segmentation network is responsible for classifying each pixel from an input image as one of the possible classes of the problem it was trained for. A thematic map is the output of this process.

SegNet [Badrinarayanan et al., 2017] has an encoder-decoder architecture (as in the Segnet block of Figure 3.5) that is followed by a pixelwise classification layer. The encoder network consists of the first 13 convolutional layers of the VGG16 network [Simonyan and Zisserman, 2014] for object classification. Each decoder layer has a corresponding encoder from which it receives max-pooling indices to perform non-linear upsampling of their input feature maps [Badrinarayanan et al., 2017]. The final decoder output serves as the input of a softmax classifier to produce the class probabilities for each pixel.

The main characteristic of the SegNet model is essentially the max-pooling indices that the encoder layers send to their corresponding decoder. This approach improves boundary delineation and reduces the number of parameters [Badrinarayanan et al., 2017]. More specifically, the max-pooling is performed with a 2×2 window and stride 2. As this operation reduces the spatial resolution and, consequently, can compromise boundary delineation, the locations of the maximum feature value in each pooling window is stored. Those indices are then used in the decoding layers in order to construct the up-sampled decoder feature maps. Max-pooling indices are less costly in terms of computational resources to store than using the feature maps themselves.

The training of the Segnet is performed with the use of a pixelwise cross-entropy loss, as in Equation 3.4.

$$Ce = -\sum_{i=1}^{C} y_i \cdot log(p_i), \tag{3.4}$$

where C is the number of classes, y_i is the ground truth and p_i the class prediction.

In order to evaluate the results of the semantic segmentation task, we selected four metrics: pixel accuracy (PA), normalized accuracy (NA), mean intersection over union (IoU) and Cohen's kappa coefficient ($Kappa$). Let n_{ij} be the number of pixels of class i predicted to belong to class j, N_c the number of classes and t_i the total number of pixels of class i . The pixel accuracy is defined as Equation 3.5.

$$PA = \frac{\sum_i n_{ii}}{\sum_i t_i}. \tag{3.5}$$

This metric represents the percentage of the total amount of pixels that were classified correctly. The normalized accuracy (NA) is defined as Equation 3.6:

$$NA = \frac{1}{N_c} \sum_i \frac{n_{ii}}{t_i}. \tag{3.6}$$

The normalized accuracy is similar to the pixel accuracy, but it normalizes the value considering the amount of pixels of each class. The IoU metric is defined as in Equation 3.7:

$$IoU = \frac{1}{N_c} \cdot \frac{\sum_i n_{ii}}{t_i + \sum_j n_{ji} - n_{ii}}. \tag{3.7}$$

The IoU metric, also known as Jaccard Index, measures the similarity between the predicted semantic segmentation map and the ground truth. It calculates the area of overlap between the predicted map and the ground truth divided by the union of both. In this case, 1 represents a perfectly predicted thematic map and 0 means there is no overlap (no pixel classified correctly). Finally, the $Kappa$ metric represents a measure of agreement between the predicted and ground truth segmentation maps. The higher the value, which can be between 0 and 1, the higher the agreement.

Regarding the existing methods for semantic segmentation, we selected SegNet because it not only has been successfully used as a base approach for methods focused on aerial data [Audebert et al., 2017; Marmanis et al., 2018], but it also presents an efficient architecture in terms of memory and computational time. This is especially important for satellite imagery, which is usually composed of very high-resolution images that can easily excess hardware limits.

3.2 Proposed Approaches

3.2.1 The Two Stage Framework

The two stage framework is presented in Figure 3.4. It was first used similarly in Dai et al. [2016], but, as explained in Chapter 2, the evaluation was highly superficial and not for remote sensing images. The pipeline of such framework is straightforward. First, a low-resolution image, from which we desire to generate a thematic map, is processed by the super-resolution network. The output from this first step is a super-resolved version of the low-resolution input that has more details and helpful features for the following network. The second step consists of inputting the super-resolved image in the semantic segmentation network. The final output, therefore, is the thematic map classifying each pixel of the super-resolved image. As the resolution and quality of this image are higher than the original input, the final thematic map should be more accurate than one generated by directly inputting the low-resolution image into the semantic segmentation network.

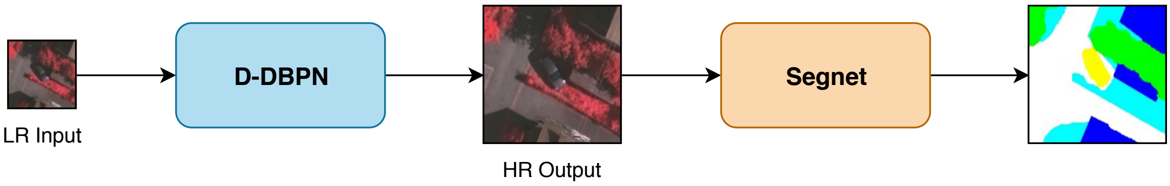


Figure 3.4: Overview of the two stage framework, which applies super-resolution on low-resolution images before sending them to a semantic segmentation network.

Although we employed a specific method for super-resolution and semantic segmentation (D-DBPN and SegNet), it is important to remark that any other one (deep learning based or not) with the same inputs and outputs could replace them according to the task, application or even preference.

Differently from Haris et al. [2018b], this framework is not trained in an end-to-end manner. Thus, the two networks (super-resolution and semantic segmentation) are trained separately. The main disadvantage of this approach is that it is not possible to use the semantic segmentation loss to bias the super-resolution network into creating an output that is more easily segmented by the other method. As verified by Haris et al. [2018b], a unified end-to-end framework can generate images that do not necessarily present the best visual characteristics for a human (when compared to applying the input to a super-resolution-only method), but that are more perceptually adequate to the vision of a machine. On the other hand, since the training of the semantic segmentation network is performed apart, any available data that does not have a corresponding thematic map can be used to train the super-resolution network. This is especially useful in the context of aerial imagery, which has less labeled data available when compared to normal images. This happens because the labeling of aerial data is usually more difficult and expensive, as many classes require the knowledge of a specialist (such as different types of vegetation). Creating data for the super-resolution network, however, is possible even without a pair of sensors with different resolutions. In this case, it is only necessary to degrade the high-resolution images and use them as input to the network. This way, with a two stage framework, we will not limit the super-resolution training only to labeled images for semantic segmentation.

For D-DBPN, we train the model for 300 epochs and randomly extract a 32×32 random patch for input from the low-resolution image on each iteration. The learning rate is initialized to $1e-4$ and is decayed by a factor of 10 at half of the total epochs. For optimization, we use Adam with 0.9 momentum and $1e-4$ weight decay.

For SegNet, we also follow the same approach that was proposed in its original paper [Badrinarayanan et al., 2017]. We train the model for 500 epochs with inputs of size 480×480. The learning rate is initialized to $1e-4$. We use Adam optimizer with 0.9 momentum and $5e-4$ weight decay. Also, in order to train SegNet in this framework, we only use available high-resolution data. On the other hand, during testing, we input the super-resolved images generated from the low-resolution ones. The motivation for this comes from the fact that in many cases, only a few amounts of data are available for training. But, in practice, we often need to perform semantic segmentation on low-resolution images. Our objective is to demonstrate that it is possible to achieve more accurate semantic segmentation results by inputting a regenerated, super-resolved version of the low-resolution images, instead of the degraded images themselves.

3.2.2 The End-to-end Framework

The end-to-end framework is capable of training the super-resolution and the semantic segmentation network at the same time. As mentioned before, this is an interesting approach as it allows the semantic segmentation network to guide the super-resolution reconstruction in a way that is more beneficial for its own vision. When using the two stage framework, the super-resolution method does not take anything into consideration apart from the network's loss and the quality of reconstruction criterion (PSNR). Thus, we can say that the reconstruction is performed aiming to improve the image characteristics for a human perspective only. The machine (semantic segmentation algorithm) vision, however, works differently from humans. Therefore, improving the visual for the human vision does not mean that it will also be ideal for the machine. By allowing the semantic segmentation loss to be also used in the training procedure together with the super-resolution loss, we are letting it bias the reconstruction procedure in a way that makes the image features more easily segmented.

Our proposed framework for this case is based on the task-driven architecture proposed in Haris et al. [2018b] and can be seen in Figure 3.5. The end-to-end framework is similar to the two stage one, considering that the super-resolution is performed before sending the result to the semantic segmentation network in both cases. It works as follows: first, the low-resolution input image is sent to the framework, where it will first be processed by the super-resolution part. The result of this process will be a super-resolved image that will be used both to calculate the $L1$ loss (super-resolution loss) and as input to the semantic segmentation part. After being processed by the SegNet, the final output of the framework will be a high-resolution thematic map made from the low-resolution input. This thematic map will also serve to calculate the semantic segmentation loss (cross-entropy). The unified loss (ξ) of the framework is calculated as in Equation 3.8, similarly to how Haris et al. [2018a] applied it to the object detection task.

$$\xi = \alpha L1(I_{HR}, SR(I_{LR}; \theta_{sr})) + \beta Ce(y_{HR}, Seg(SR(I_{LR}; \theta_{sr}); \theta_{seg})), \tag{3.8}$$

where $L1(.)$ represents the super-resolution loss, as in Equation 3.3, and $Ce(.)$ the cross-entropy loss for semantic segmentation, as in Equation 3.4. I_{HR} and I_{LR} represent, respectively, the high-resolution ground truth image and the low-resolution input. y_{HR} is the ground-truth thematic map. $SR(.)$ and $Seg(.)$ are, respectively, the super-resolution and semantic segmentation networks with their set of parameters θ. Finally, α and β are pre-defined values that represent the balance between the super-resolution and semantic segmentation losses.

The definition of the α and β values is the key to defining how biased the outputs will be for human or machine perception. With an α value higher than β, the network will prioritize the super-resolution reconstruction over the result of the semantic segmentation. However, by setting a β value higher than α, the framework will penalize more the semantic segmentation error and consider less how the image reconstruction

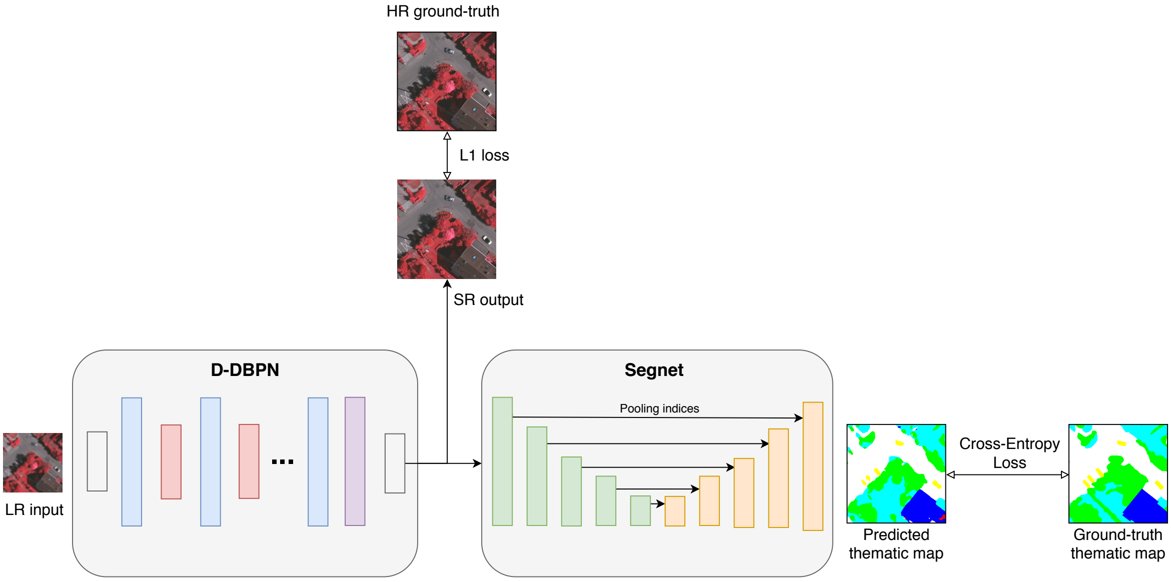


Figure 3.5: Overview of the end-to-end framework, which trains the super-resolution and semantic segmentation networks at the same time.

is being performed. This means that the SegNet will be able to conduct the training of D-DBPN in a way it is better for itself to see the relevant features.

This framework is trained for 300 epochs with inputs of size 480 × 480. The learning rate is initialized to $1e-5$ and is decayed by a factor of 10 at half of the total epochs. Each individual network inside the framework (D-DBPN and SegNet) is optimized under the same conditions as they do in the two stage framework.

Chapter 4

Experiments and Results

4.1 Experimental Setup

The objective of the experiments is to evaluate how well can super-resolution be used to improve semantic segmentation by using low-resolution images as input. We simulate this situation by evaluating the effectiveness of the frameworks presented in Chapter 3 with images in different levels of resolution degradation. Subsection 4.1.1 presents the datasets we applied in the experiments. In Subsection 4.1.2, we describe implementation details of the experimental protocol.

4.1.1 Datasets

In order to evaluate our framework, we selected three distinct remote sensing datasets. The first one is the Brazilian Coffee Scenes Dataset [Penatti et al., 2015], which is an agricultural dataset composed of scenes containing coffee and non-coffee areas. The second one is the Vaihingen dataset, provided by the International Society for Photogrammetry and Remote Sensing (ISPRS) Commission for the 2D Semantic Labeling Contest, which contains urban scenes with six different pixel classes. The third one is the 2014 IEEE GRSS Data Fusion Contest dataset, that also contains urban scenes and seven thematic classes.

1. **Coffee Dataset**: this dataset contains images from three different Brazilian cities from the state of Minas Gerais: Monte Santo, Guaranésia and Guaxupé. They are composed of green, red, and near-infrared bands with over 6000 × 10000 pixels each. Although having only 2 classes (coffee and non-coffee), this is a challenging dataset, as noted by Nogueira et al. [2019], since it contains

high intraclass variance, scenes with distinct plant ages and images with spectral distortions caused by shadows.

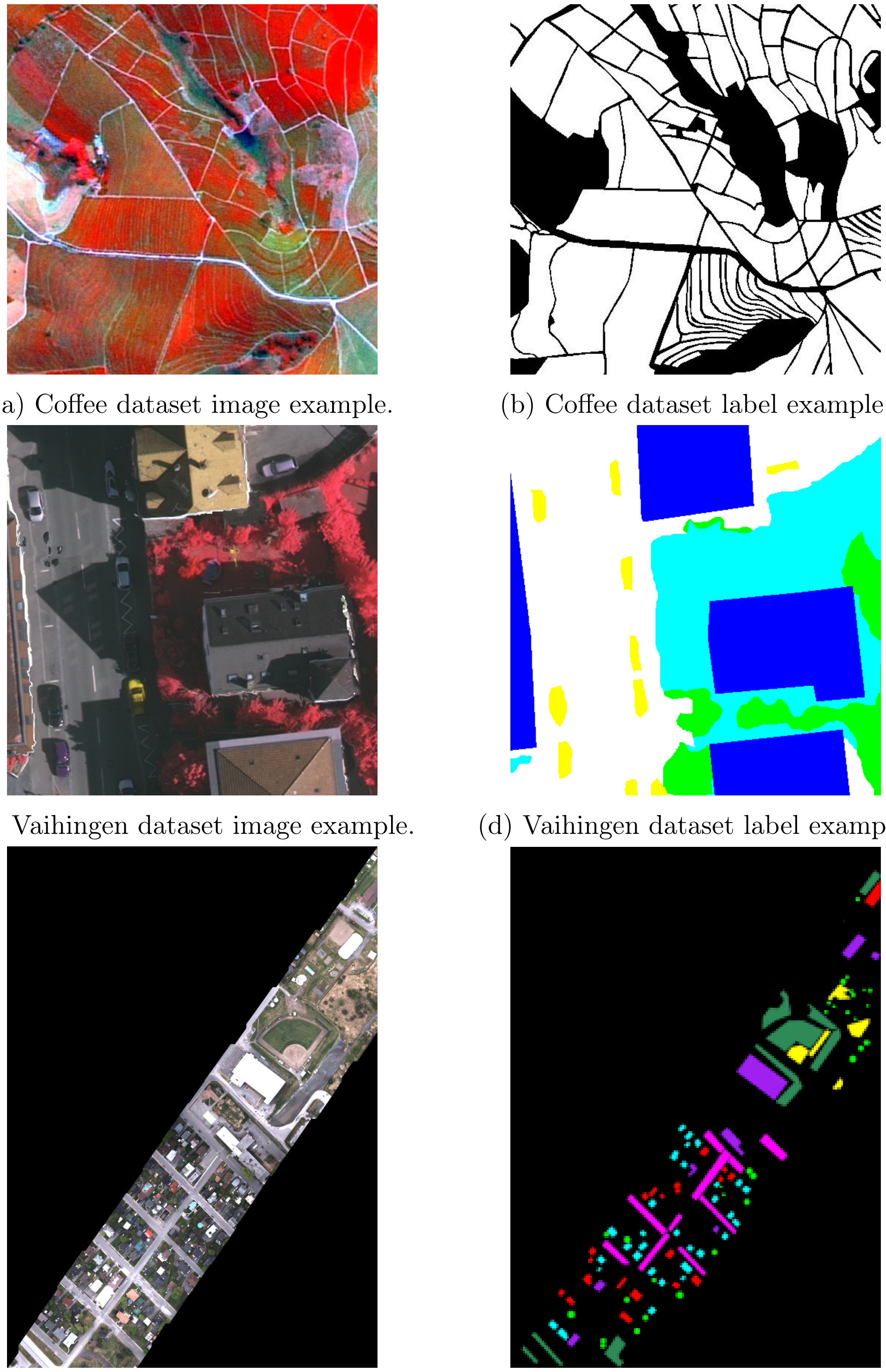
2. **Vaihingen Dataset**: this dataset contains 33 high-resolution images and six different thematic labels: impervious surfaces, building, low vegetation, tree, car, clutter/background. Similarly to the coffee dataset (although with a different order), the images are composed of near-infrared, red and green bands. The ground sampling distance is $9cm$. From all the images, only 16 of them have ground-truth available for the semantic segmentation task. As mentioned in Chapter 3, the two stage framework pipeline allows us to train the super-resolution network with all 33 images, while only using the 16 labeled ones for the semantic segmentation task.

3. **Thetford Dataset** (2014 IEEE GRSS Data Fusion Contest Dataset): this dataset contains one image (divided into two RGB sub-images) from an urban area near Thetford Mines (Quebec, Canada) and seven thematic labels: trees, vegetation, road, bare soil, red roof, gray roof, and concrete roof, along with the unclassified pixels. This dataset presents a spatial resolution of approximately $20cm$.

Figure 4.1 shows the training image from the Thetford dataset and one example of a crop of an image from both Coffee and Vaihingen datasets.

The selected datasets fit well for our purpose since they present different characteristics that can be explored by the networks. The coffee dataset, for example, requires a lot of texture information to be able to distinguish coffee crops from non-coffee areas, while the urban datasets contain small objects (such as cars) that need to be visible enough for the network to classify their pixels correctly.

For the coffee dataset, we employed a protocol in which we train the networks on the images of two cities and test them on the remaining city. Thus, we train the models on Montesanto and Guaxupé and test on Guaranésia. Then, we train on Guaxupé and Guaranésia and test on Montesanto. Finally, we train on Guaranésia and Montesanto and test on Guaxupé. Later, the results for this dataset will be reported in terms of the mean and standard deviation of these three cases. By separating the images of a whole distinct city for the test, we simulate a real-world scenario in which we have high-resolution images taken from different cities, but need to apply the semantic segmentation on low-resolution data from a new location. This case could not be reproduced with fidelity by simply selecting for test random crops of all the available data.



(a) Coffee dataset image example.

(b) Coffee dataset label example.

(c) Vaihingen dataset image example.

(d) Vaihingen dataset label example.

(e) Thetford dataset training image.

(f) Thetford dataset training label.

Figure 4.1: Example of an image and its semantic segmentation label from the three chosen datasets.

Labeled ground truth is provided for only one part of the Vaihingen dataset since the remaining scenes were not released and require submission to the benchmark test organizers to be evaluated. Therefore, we trained and tested our framework using only the publicly available images. We applied the same division of the data as in Nogueira et al. [2019]: areas 11, 15, 28, 30 and 34 are used as test, while the remaining areas are seen during training. We also point out that even though this dataset contains six classes, we excluded from the results the clutter/background (red label) class. The reason for this is that the class represents less than 1% of the dataset (being highly mislabeled even with high-resolution inputs), since it is designated to unclassified or rejected objects on the scene, and it is also not considered in some similar works, such as the ones proposed in Nogueira et al. [2019] and Maggiori et al. [2017].

For the Thetford dataset, we use the same parts of the image selected by the contest for training and test. The original dataset contains seven classes, but one of them (bare soil, yellow label) is only present in the training part of the full data. As this sub-image is also used to train D-DBPN, it would not be fair to compare the performance with bicubic interpolation, since we would be applying super-resolution on the same data used to train. Thus, we do not considerate the bare soil class in our results.

4.1.2 Implementation Details

We evaluate our frameworks under two scaling factors of degradation: 4× and 8×. Our objective is to compare how much the super-resolution network can help in each case. Intuitively speaking, low amounts of degradation (×2, for example) should not present enough differences from the low-resolution image to compensate the need for a super-resolution network. On the other hand, high amounts of degradation (such as more than 8×) should make it almost impossible to recover enough information when creating the super-resolved image.

As described in Chapter 3, the evaluation for super-resolution is different from semantic segmentation. We evaluate the quality of regenerated images in terms of Peak Signal-to-Noise Ratio (PSNR), which is the default metric for most of the super-resolution methods. Since two of the datasets we selected are not RGB, we evaluate the PSNR over all the three channels of the inputs (instead of only the Y channel of YCbCr, which is the usual protocol for super-resolution on RGB images). For the semantic segmentation, we used the four metrics described in Chapter 3: pixel accuracy (acc), normalized accuracy ($norm.acc$), mean intersection over union (IoU) and Cohen's kappa coefficient ($Kappa$). These are the final values we use to evaluate how much the super-resolution interfered on the result, as the PSNR is just a measure of how well an image was reconstructed.

We applied a similar experimental protocol for each one of the datasets. First of all, we divide the training and testing high-resolution images in crops of size 480×480, from which we create the low-resolution inputs of size 120×120 and 60×60 for, respectively, $4\times$ and $8\times$ up-scaling factors. The choice for this dimension comes from the fact that the original Segnet paper [Badrinarayanan et al., 2017] uses inputs of size 360×480. In order to create the low-resolution images, we follow the same approach used by the track in which D-DBPN won the NTIRE2018 challenge and that was also used by Haris et al. [2018b] (on their first experiment), Dai et al. [2016] and Ferdous et al. [2019], which is a bicubic kernel. Thus, we apply bicubic interpolation on the high-resolution image with the desired down-scaling factor.

The weights of D-DBPN are initialized with the pre-trained model provided by the original Github repository of the paper [Haris et al., 2018a]. Similarly, Segnet is fine-tuned with the VGG16 trained weights for image classification. For the end-to-end framework, we also initialize each one of these corresponding blocks in the same way.

For the two stage framework, following the pipeline explained in Chapter 3, we start off by training the super-resolution network, D-DBPN, for the desired up-scaling factor using the pairs of original high-resolution data and the generated low-resolution images. During this stage, as mentioned before, we can also use data that does not contain a corresponding thematic map in order to improve the super-resolution training. Regardless of this step, we also need to train the semantic segmentation network with the available high-resolution data and the thematic maps. During these two stages, it is crucial to make sure the images that are going to be used as test on the semantic segmentation task will not be included in the training set of the super-resolution network. This is important since we only want to apply super-resolution on the images that none of the networks saw during training, or else we would be creating a bias towards the reconstruction of the low-resolution image. This procedure works as a simulation of a real-world scenario, in which we have already trained the models with the few available high-resolution data and now we need to apply the semantic segmentation on a new low-resolution image. After training both networks separately, the final evaluation is performed on the output of the semantic segmentation network.

Since the training of the end-to-end framework is not performed in two parts, the super-resolution and semantic segmentation networks are trained together in a single step. This means that unlabeled data for semantic segmentation will not be used in the framework at all. However, the lesser amount of images used to train the super-resolution network is compensated by the fact that this network is trained to create images that are especially appropriate for the semantic segmentation task. While the two stage framework simulates a situation in which we train the semantic segmentation network with high-resolution data and test on reconstructed data generated by a super-resolution network, the end-to-end framework simulates the whole process of training and testing already expecting a low-resolution image as input. Therefore, this framework is less versatile than the previous (detaching the semantic segmentation block will not allow it to perform well in high-resolution data), but it is more powerful on low-resolution images.

We experimented many different losses configurations for the end-to-end framework by changing the α and β values of Equation 3.8. As our objective is to improve semantic segmentation results, we aim for higher β values. We tested the framework on the Vaihingen dataset with α values from the set $\{0.001, 0.1, 1\}$, and β values from the set $\{1, 10, 100, 1000, 10000, 100000\}$. As a general rule, we observed that lower β values achieved results that were similar to the two stage framework, since the semantic segmentation loss, in these cases, does not cause enough influence in the reconstruction loss. We also observed that the higher the β values, the higher was the number of artifacts created in the reconstructed image. Under these circumstances, the best results were achieved with the 0.1/1000 configuration for α and β, respectively. Thus, the results reported next for the end-to-end framework are all using this same configuration.

4.2 Results and Discussion

We conducted an extensive series of experiments in order to answer the following research questions: (1) How effective is deep-based super-resolution to different levels of degradation for remote sensing semantic segmentation tasks? (2) How deep-based super-resolution compares to classical unsupervised interpolation? (3) Is deep-based super-resolution able to reconstruct small objects and, consequently, contribute to semantic segmentation improvement?

4.2.1 Effectiveness to different levels of degradation

In this subsection, we present the results that allow the evaluation of the effectiveness of the two frameworks with 4× and 8× degradation factor. Table 4.1 presents the performance of the two stage framework.

The results show, for all datasets, that image resolution has a high impact on semantic targeting results. In general, the lower the resolution, the worse the result. However, the impact of the degradation rate impacts differently for each dataset.

Table 4.1: Semantic Segmentation Performance of the Two Stage Framework for Different Degradation Factors

Dataset	**Deg.**	**Acc**	**Norm. acc**	**IoU**	**Kappa**
Coffee	8×	0.7637 ± 0.0119	0.7204 ± 0.0300	0.5817 ± 0.0305	0.4631 ± 0.0471
	4×	0.8029 ± 0.0050	0.7722 ± 0.0036	0.6454 ± 0.0060	0.5626 ± 0.0091
	1×	0.8330 ± 0.0136	0.8168 ± 0.0091	0.6972 ± 0.0175	0.6390 ± 0.0241
Vaihingen	8×	0.7447	0.5931	0.4762	0.6621
	4×	0.7912	0.6369	0.5256	0.7234
	1×	0.8479	0.6833	0.5909	0.7984
Thetford	8×	0.5444	0.6000	0.2916	0.4065
	4×	0.7178	0.6665	0.4268	0.5897
	1×	0.8452	0.8184	0.6463	0.7636

Concerning coffee, the segmentation quality loss is relatively low for all metrics, ranging, for example, from mean 0.81 to 0.72 in the case of normalized accuracy from the original high-resolution image to the same image with 8× resolution degradation factor. It is an indication that for cropping, the use of deep-based super-resolution can improve the results. In the case of the urban datasets, the impact of the loss of resolution was greater than for coffee crops. Regarding the Thetford dataset, in particular, the normalized accuracy was reduced from 0.81 to 0.60 for 8× degradation. For the Vaihingen dataset, the normalized accuracy was reduced from 0.68 to 0.59. The main explanation for the effect is that the Coffee dataset has only two classes and, in general, the coffee crops are relatively large areas. They also depend more on the texture than in the shape. In the case of the urban scenes, the accuracy was reduced mainly due to classes such as trees and cars that are composed of small regions that are difficult to recover given the strong loss of information.

Even though the urban datasets were more affected by the degradation than the coffee dataset, the difference for Thetford was higher than for Vaihingen. This can be explained by the high amount of data that is available for the Vaihingen dataset, especially to train D-DBPN. As mentioned before, some images from the dataset were not labeled for semantic segmentation, but were used to train the super-resolution network. The consequence for this is that the quality of the reconstructed images is much higher compared to Thetford (which contains a small quantity of training data), thus helping more the semantic segmentation task. This indicates that deep-based super-resolution can increase the semantic segmentation results relatively close to a native high-resolution data given enough training.

Table 4.2 shows the semantic segmentation results of the end-to-end framework.

Table 4.2: Semantic Segmentation Performance of the End-to-End Framework for Different Degradation Factors

Dataset	**Deg.**	**Acc**	**Norm. acc**	**IoU**	**Kappa**
Coffee	8×	0.8003 ± 0.0259	0.7784 ± 0.0245	0.6477 ± 0.0343	0.5653 ± 0.0521
	4×	0.8205 ± 0.0121	0.8093 ± 0.0092	0.6807 ± 0.0149	0.6162 ± 0.0207
	1×	0.8330 ± 0.0136	0.8168 ± 0.0091	0.6972 ± 0.0175	0.6390 ± 0.0241
Vaihingen	8×	0.8288	0.6625	0.5654	0.7730
	4×	0.8293	0.6631	0.5697	0.7738
	1×	0.8479	0.6833	0.5909	0.7984
Thetford	8×	0.8605	0.8564	0.6986	0.7881
	4×	0.8733	0.8417	0.7117	0.7997
	1×	0.8452	0.8184	0.6463	0.7636

It is possible to see that the impact of 8× degradation factor while using the end-to-end approach is not as considerable compared to 4× as in the two stage framework. This is true especially for the urban datasets. The Vaihingen dataset results for the two degradation factors are quite close. In the Thetford dataset, the normalized accuracy was even higher when inputting 8× degraded images (mostly due to class balance factors), while the remaining metrics also stood close. This indicates that the end-to-end framework is more capable of dealing with higher degradation factors without losing too much semantic segmentation accuracy. This is due to the fact that this framework can change the reconstructed image with information that is more easily discernible for the semantic segmentation task. When relying only on the super-resolution loss (as in the two stage framework), there is no assurance that similar textures (such as trees and vegetation, building and impervious surface) will be reconstructed in a way that highlights the differences among them. By letting the semantic segmentation task guide the super-resolution, we are allowing this highlighting to occur automatically. Another important reason that makes the end-to-end framework perform better is that it is trained with low-resolution data as input, while the Segnet of the two stage framework is trained with high-resolution data. Therefore, the difference in the degrading factors impacts more a network trained with high-resolution data only than a network trained with that specific degradation as input.

We can also see that even the difference to native high-resolution data (1× degradation) is smaller in the end-to-end framework. Taking the same example of the previous experiment, for the Coffee dataset, the normalized accuracy was reduced from mean 0.81 (for native high-resolution data) to 0.77 with 8× degradation. This is a 4% difference from the original high-resolution data to the restored one with the highest degradation factor. The difference for the two stage framework in the same condition was 9%. Considering the *IoU* metric, the end-to-end framework achieved 0.56 with 8× degradation on the Vaihingen dataset (0.59 on native high-resolution data). In the same case, the two stage framework could only achieve 0.47, which is way more distant compared to native high-resolution data. The most interesting and noticeable change, though, is in regard to the Thetford dataset. The two stage framework lost 21% normalized accuracy and 35% *IoU* with 8× super-resolution compared to the SegNet trained on native high-resolution inputs. The end-to-end framework, on the other hand, actually managed to achieve better results with low-resolution images than the semantic segmentation trained on high-resolution data. One of the reasons that made this happen is the low amount of training data for this dataset. The lack of training images compromised the SegNet to differentiate similar classes and deal with the intra-class variance. However, as the end-to-end framework allows the image to be changed in order to help the semantic segmentation (which makes it easier to differentiate similar classes), and considering that the low-resolution aspect also diminishes the intra-class variance, the results of the framework ended up being better than expected. With enough training data, though, the results for a semantic segmentation network trained with high-resolution images would probably follow the same pattern as the Vaihingen dataset (i.e., semantic segmentation with high-resolution data performing better and low-resolution data applied to the framework closely behind).

What the results show is that the end-to-end framework is more effective than applying super-resolution as a pre-processing step for semantic segmentation trained with high-resolution data. This was an expected conclusion, since the semantic segmentation network of the two stage framework is trained with high-resolution data and, therefore, is not being tested with the same resolution. The end-to-end framework, on the other hand, is trained and tested expecting the same resolution as input. This does not change the fact that super-resolution can indeed improve the results for low-resolution data when the training is performed on high-resolution images. The results also show that the end-to-end framework is able to reconstruct the low-resolution image taking into consideration the vision of the semantic segmentation algorithm, rather than trying to improve the image for the human perspective. Thus, the end-to-end framework was able to achieve results that are closer to original high-resolution data or even better depending on the amount of training data.

Table 4.3: Comparison between the performance of bicubic interpolation and super-resolution in the two stage framework and the end-to-end framework.

Dataset	Deg.	Method	PSNR (dB)	Norm. acc	Kappa	IoU
Coffee	4×	Bicubic	25.2097 ± 0.7974	0.5669 ± 0.0217	0.1610 ± 0.0507	0.4015 ± 0.0275
		Two stage	27.2782 ± 1.1140	0.7722 ± 0.0036	0.5626 ± 0.0091	0.6454 ± 0.0060
		End-to-end	26.055 ± 0.3692	0.8093 ± 0.0092	0.6162 ± 0.0207	0.6807 ± 0.0149
	8×	Bicubic	21.6604 ± 0.6941	0.5012 ± 0.0007	0.0032 ± 0.0019	0.3173 ± 0.0039
		Two stage	22.8333 ± 1.1822	0.7204 ± 0.0300	0.4631 ± 0.0471	0.5817 ± 0.0305
		End-to-end	21.2722 ± 1.3491	0.7784 ± 0.0245	0.5653 ± 0.0521	0.6477 ± 0.0343
Vaihingen	4×	Bicubic	28.7458	0.5741	0.6417	0.4526
		Two stage	31.1974	0.6369	0.7234	0.5256
		End-to-end	26.3690	0.6631	0.7738	0.5697
	8×	Bicubic	25.3886	0.4747	0.5281	0.3449
		Two stage	27.4540	0.5931	0.6621	0.4762
		End-to-end	22.6199	0.6625	0.7730	0.5654
Thetford	4×	Bicubic	26.8292	0.5776	0.4271	0.2906
		Two stage	31.0294	0.6665	0.5897	0.4268
		End-to-end	29.8188	0.8417	0.7997	0.7117
	8×	Bicubic	23.3354	0.4660	0.1672	0.1408
		Two stage	26.3173	0.6000	0.4065	0.2916
		End-to-end	25.5920	0.8564	0.7881	0.6986

4.2.2 Comparison to bicubic interpolation

Table 4.3 presents the results for semantic segmentation by using bicubic interpolation and super-resolution (both as a pre-processing step and in the end-to-end approach). It also reports the reconstruction rate with PSNR. As the table shows, the use of super-resolution improved the results of all the metrics for all datasets and degradation factors. This is especially true for higher degradation factors (8×), since the loss of information is more considerable and, thus, deep-based super-resolution can learn to recover the details much better than a simple interpolation.

The big difference in the semantic segmentation metrics shows that super-resolution allowed the network to predict with more precision the classes of each dataset when compared to interpolated low-resolution inputs. This is another clear evidence of the capability of super-resolution to recover important visual details for a semantic segmentation algorithm. The most impressive improvement can be noted in the coffee dataset results. Using bicubic interpolation, the normalized accuracy is close do 50%, the $Kappa$ values are close to zero and the IoU is way smaller than employing the frameworks with super-resolution. This happened because the loss of texture information was so severe that SegNet was incapable of detecting most of the coffee areas, thus classifying almost 100% of the images as non-coffee. Examples of this situation can be seen in Figure 4.2. Employing super-resolution, on the other hand, allowed the coffee areas to be segmented with much more precision.

Another important improvement can be noted for the Thetford dataset. Being able to increase the performance with deep-based super-resolution even when it contains a small amount of training data (a bad scenario for a deep learning algorithm), shows that our frameworks are more reliable than interpolation.

Regarding the reconstruction, the PSNR is higher when applying D-DBPN directly as an up-scaling method (as in the two stage framework) instead of a simple bicubic interpolation. This means that the super-resolved output contains more visually appealing, high-frequency details than an interpolated image. However, the PSNR in the end-to-end approach does not follow the same pattern. Even though the reconstruction metric is not as high as the other options in some cases, the semantic segmentation results are better. That happens because the supervision of the semantic segmentation network in the training of the super-resolution method allows the framework to change the visual characteristics of the reconstructed image. This makes the PSNR drop since the super-resolution output will present details that are inexistent in the ground-truth high-resolution image, but those details are exactly what makes the performance of the SegNet improve.

Considering the relationship between PSNR (reconstruction metric) and the semantic segmentation metrics, there are two ways of analyzing the results. When we consider only the two stage framework, the semantic segmentation performance is better with higher PSNR values. Thus, we can verify that this metric correlates well with how better a reconstructed image can be segmented. However, when we look at the results of the end-to-end framework, we can verify that lower PSNR values do not necessarily imply bad semantic segmentation performance. It all depends on how the reconstruction method is performed. If the objective of the framework is to reconstruct a low-resolution image based on the human perspective (as in the two stage framework), the PSNR is a valid alternative to create better semantic segmentation results. On the other hand, letting the machine conduct the reconstruction may lead to better semantic segmentation results, even if it presents worse reconstruction outputs visually speaking.

4.2.3 Robustness to small object segmentation

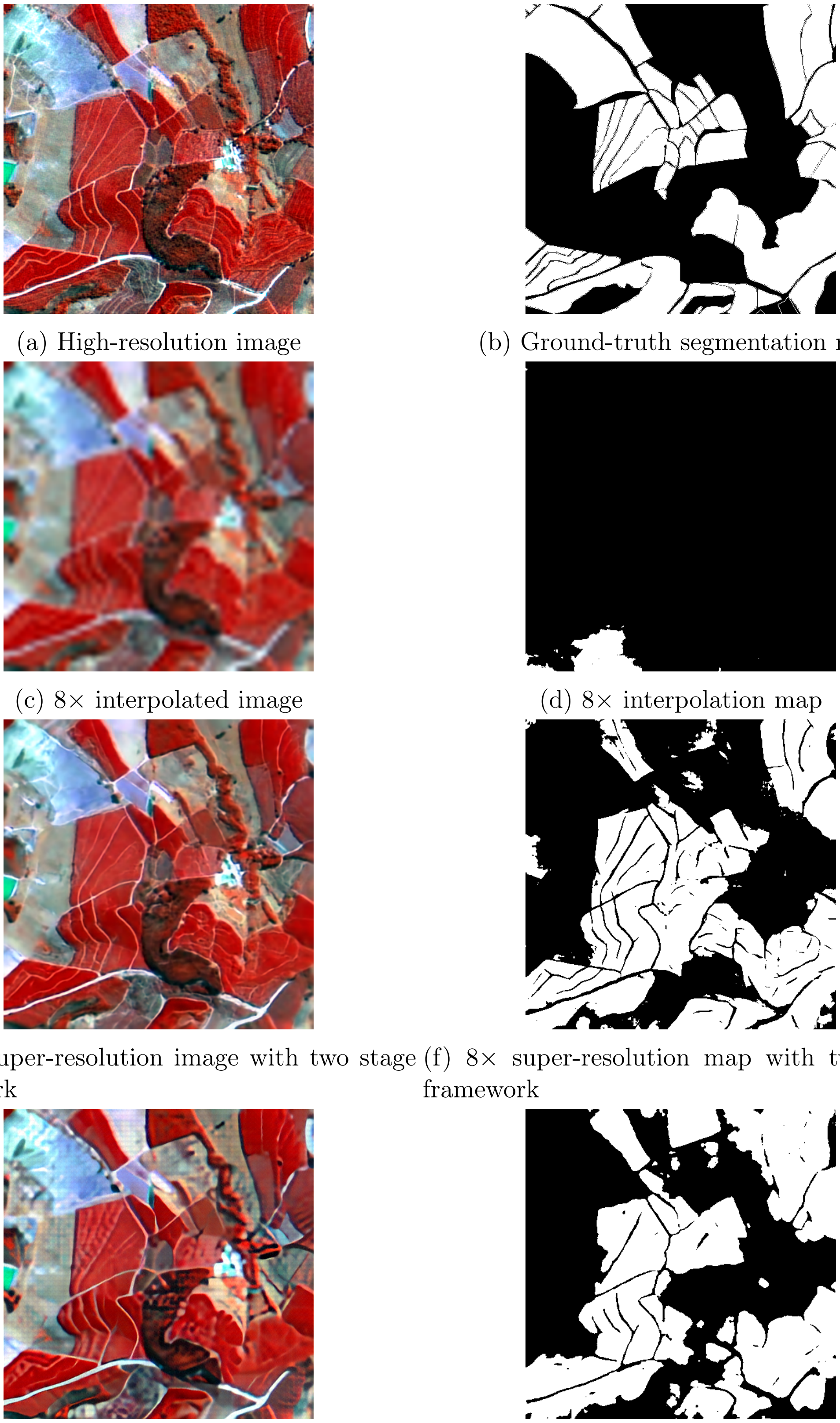
In order to verify the effectiveness of the two frameworks in the segmentation of small objects, we analyzed the results obtained by class for the two urban datasets: Vaihingen and Thetford. This can be seen in the confusion matrix of Figures 4.3, 4.4, 4.7 and 4.8. Visual results for super-resolution and semantic segmentation are shown in Figures 4.5, 4.6, 4.9 and 4.10.

Comparing the confusion matrix for 4× and 8× super-resolution, it is possible to observe indications of the same conclusions obtained in the previous research questions. The first observation is that higher degradation factors impact more in the semantic segmentation results, but the end-to-end framework is more capable of dealing with such problem than the two stage one. Again, this happens especially because it is trained and tested on images of the same input resolution, differently from the two stage framework pipeline. For both datasets, the end-to-end framework managed to stay close to the accuracy of high-resolution inputs even under 8× degradation. It is also possible to see that the use of low-resolution data (with bicubic interpolation) compromises a lot the performance of semantic segmentation. The trees on Thetford dataset were almost 100% mislabeled even with only 4× degradation. For the Vaihingen dataset, the use of bicubic interpolation prejudiced the most the car class. In Figure 4.6 we can see an example of a segmentation that missed most of the car information due to the low-resolution representation, but that was successfully recovered with the use of both frameworks.

For the Vaihingen dataset, it is possible to verify in Figure 4.4 that super-resolution greatly improved the segmentation of the car class: for 8× degradation, bicubic up-sampled inputs could predict correctly only 19% of the car pixels, while super-resolved inputs in the two stage framework increased this value to 58% and the end-to-end framework improved this result further to 65% (the SegNet trained on high-resolution data achieved 69%). This confirms that super-resolution and both frameworks are capable of making more discernible (for a machine) objects that are too small in a low-resolution representation. The results also present a great improvement for the building class, which was highly mislabeled as impervious surfaces by the low-resolution representation, but that was better segmented on super-resolved inputs, increasing the value from 31% to 68% and 89% for the two stage and end-to-end frameworks, respectively, with 8× degradation. In this dataset, the two stage framework lost to interpolation only on the tree class (a 5% difference on 8× degradation). However, looking at the low values of accuracy and IoU presented in Table 4.3, the reason for this is that the network simply classified a higher amount of pixels as tree, what increases the chances of predicting this class, but causes many cases of mislabeling for the other labels. This is true especially for low vegetation, which was mislabeled as tree only 9.2% of the time on super-resolved inputs of the two stage framework, but 16% on bicubic up-sampled images.

As for the Thetford dataset, applying semantic segmentation on high-resolution data resulted in accurate predictions, which is confirmed by the high diagonal numbers in Figure 4.7 and 4.8. Super-resolution with the two stage framework and bicubic interpolation, however, mislabeled trees, roads and grey roofs in many cases. The interesting aspect, though, comes from the results of the end-to-end framework. As discussed before, the semantic segmentation metrics for this approach were higher than even native high-resolution data. This is confirmed by the values in the confusion matrices: considering the 8× degradation factor, the end-to-end framework considerably surpassed the accuracy of high-resolution data for the tree, red roof, grey roof, and concrete roof classes, while considerably staying behind only on the vegetation class. With more training data, though, the semantic segmentation network applied to high-resolution data could surpass the framework without the need for artificially generating different textures in the images.

Finally, by observing the visual results of the reconstructed images from the end-to-end framework in Figures 4.5, 4.6, 4.9 and 4.10, it is possible to see the different textures employed by the semantic segmentation network that helped it to more accurately classify the pixels. As mentioned before, this is also the reason why the PSNR values for these cases are lower than even bicubic interpolation, but without prejudicing the semantic segmentation performance.



(a) High-resolution image

(b) Ground-truth segmentation map

(c) 8× interpolated image

(d) 8× interpolation map

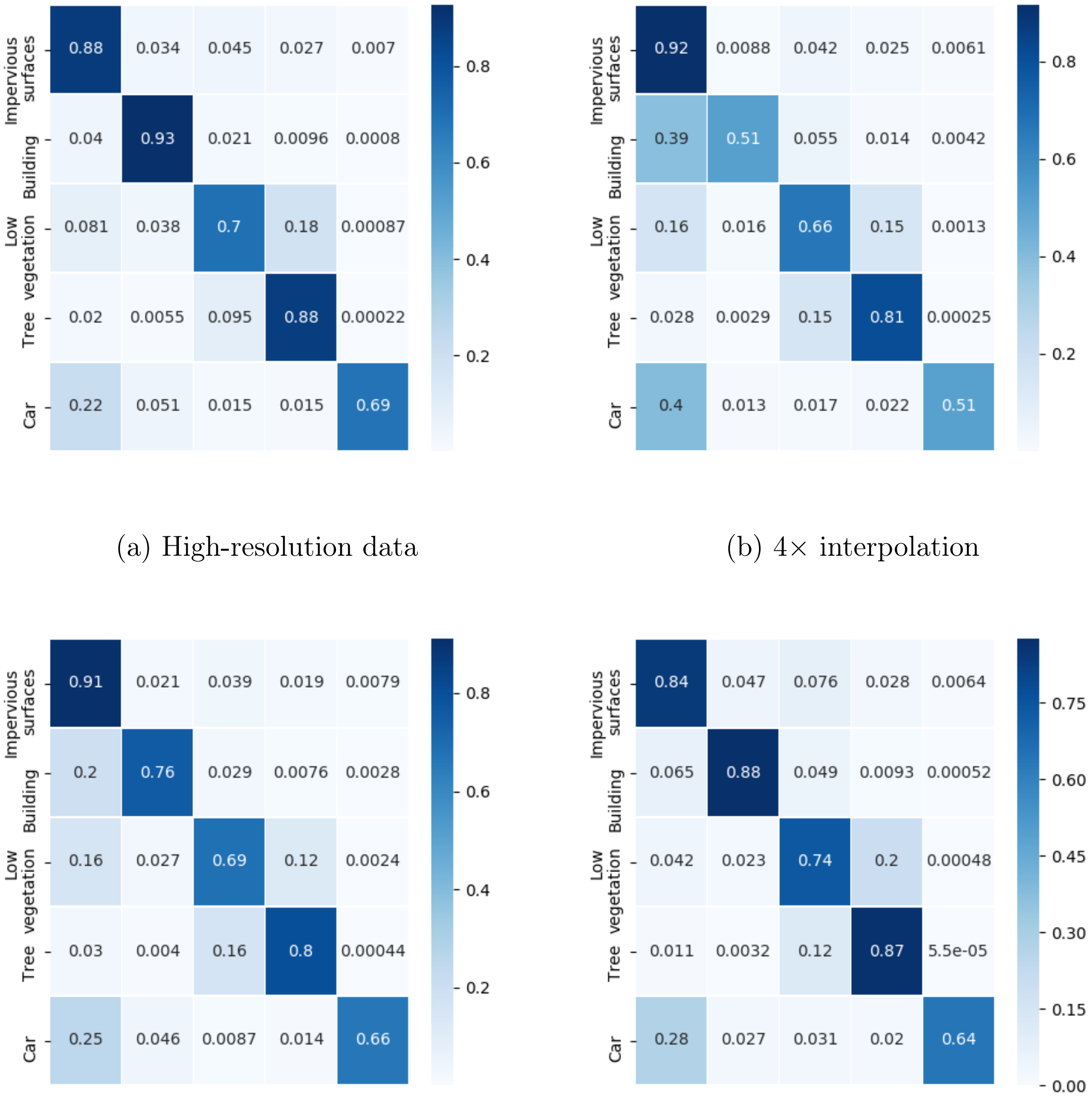
(e) 8× super-resolution image with two stage framework

(f) 8× super-resolution map with two stage framework

(g) 8× super-resolution image with end-to-end framework

(h) 8× super-resolution map with end-to-end framework

Figure 4.2: Example results for the Coffee dataset with 8× up-scaling factor.



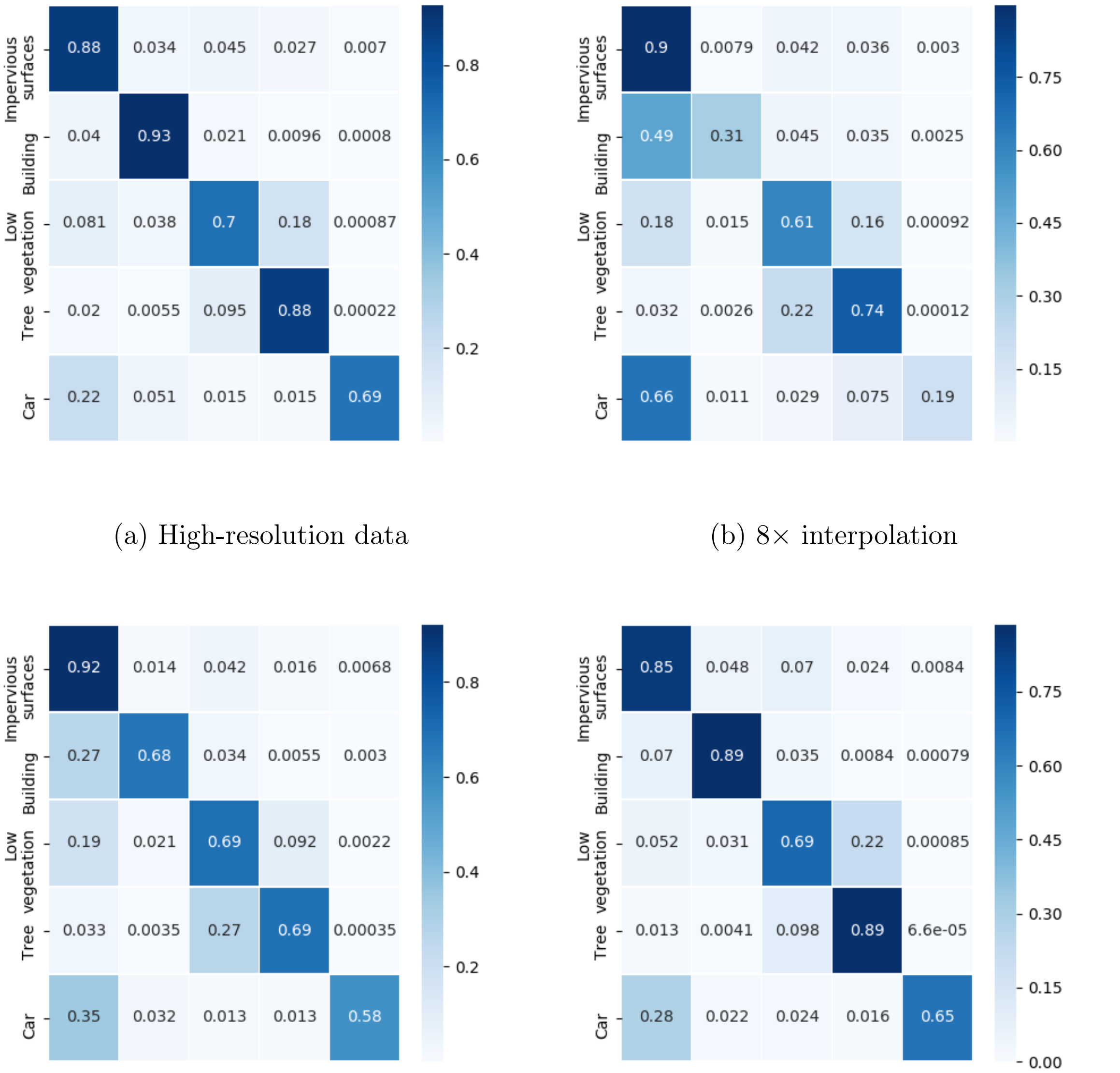
(a) High-resolution data

(b) 4× interpolation

(c) 4× super-resolution with two stage framework

(d) 4× super-resolution with end-to-end framework

Figure 4.3: Accuracy heatmaps for semantic segmentation on the Vaihingen dataset with 4× up-scaling.



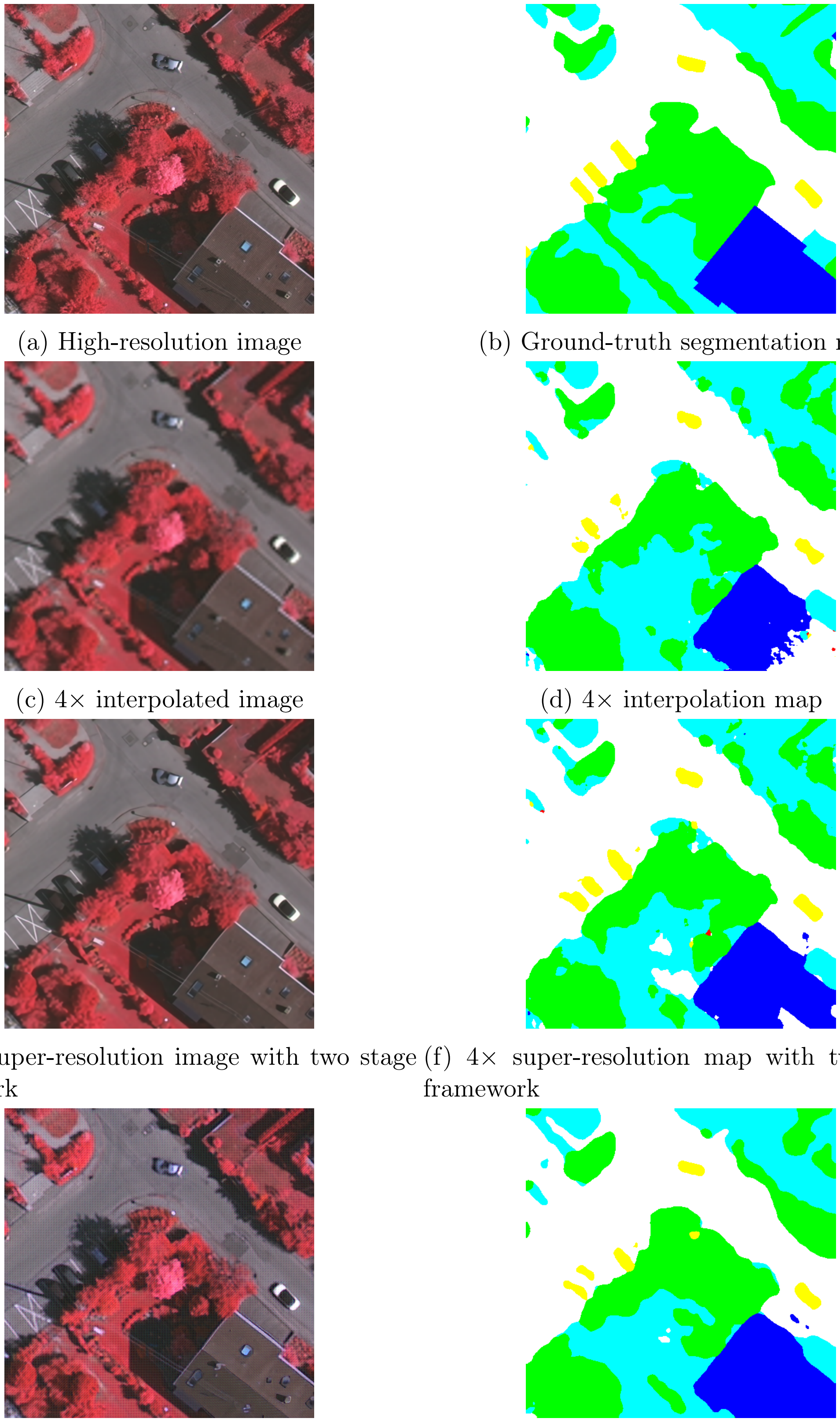
(a) High-resolution data

(b) 8× interpolation

(c) 8× super-resolution with two stage framework

(d) 8× super-resolution with end-to-end framework

Figure 4.4: Accuracy heatmaps for semantic segmentation on the Vaihingen dataset with 8× up-scaling.



(a) High-resolution image

(b) Ground-truth segmentation map

(c) 4× interpolated image

(d) 4× interpolation map

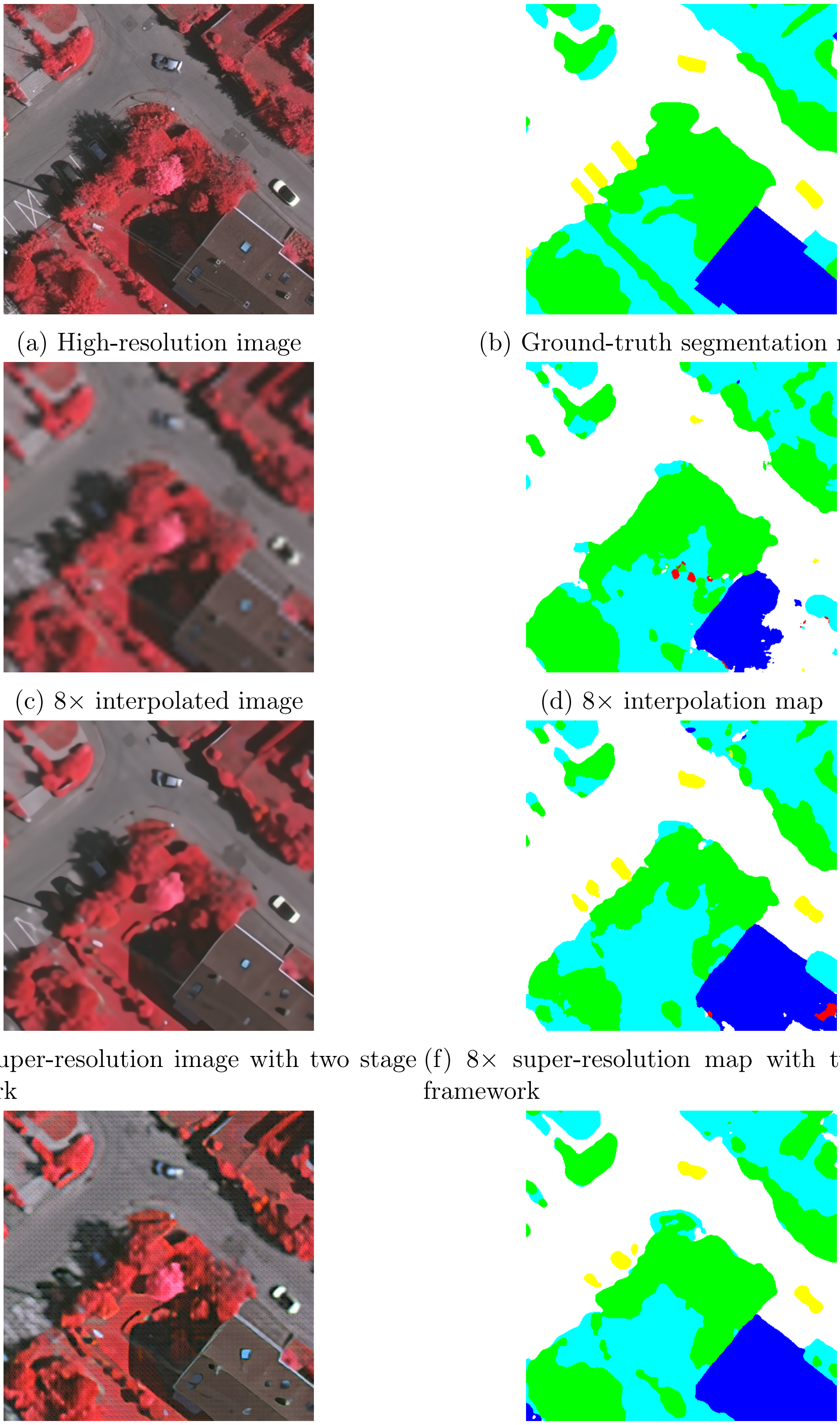
(e) 4× super-resolution image with two stage framework

(f) 4× super-resolution map with two stage framework

(g) 4× super-resolution image with end-to-end framework

(h) 4× super-resolution map with end-to-end framework

Figure 4.5: Example results for the Vaihingen dataset with 4× up-scaling factor.



(a) High-resolution image

(b) Ground-truth segmentation map

(c) 8× interpolated image

(d) 8× interpolation map

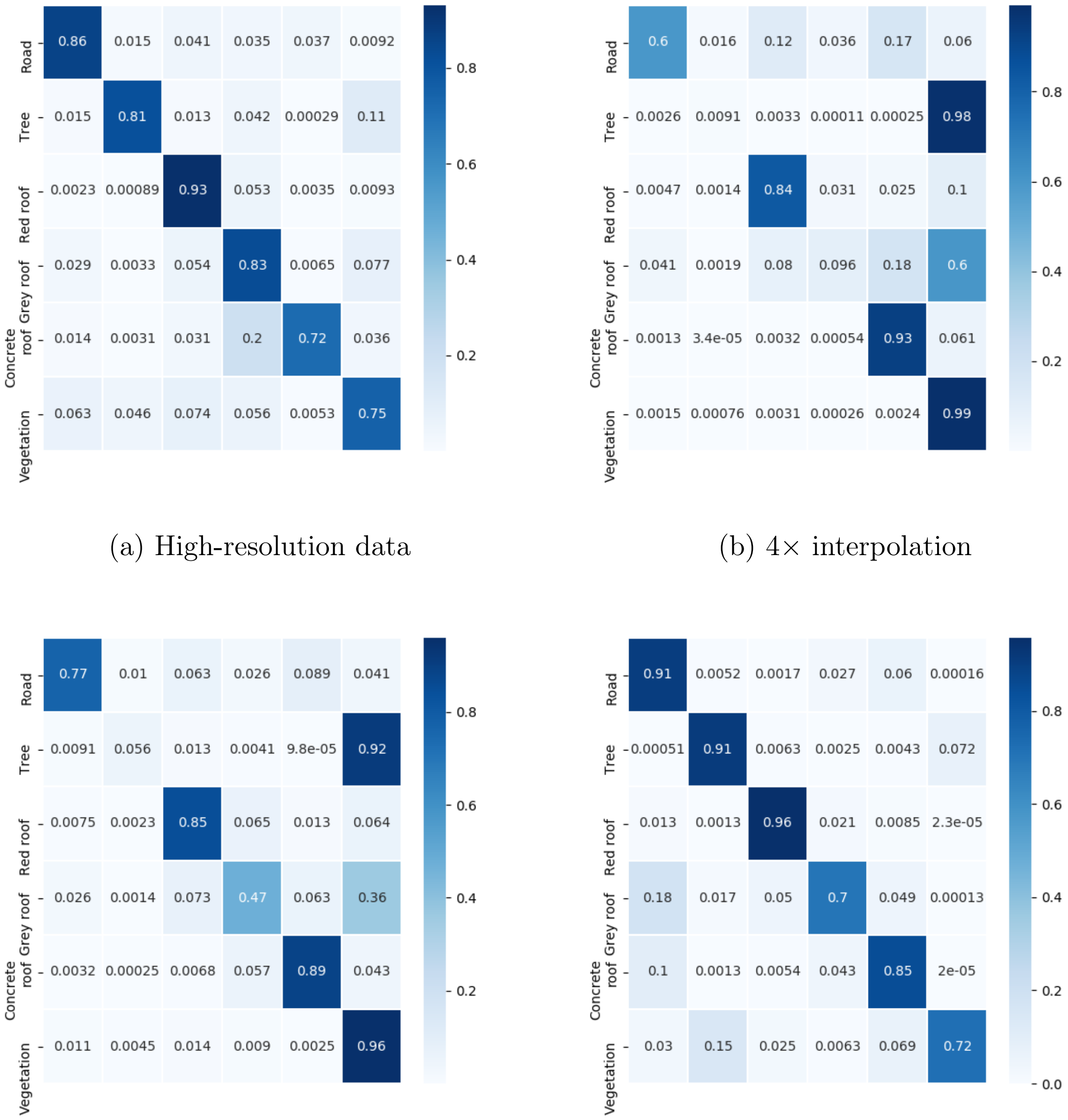
(e) 8× super-resolution image with two stage framework

(f) 8× super-resolution map with two stage framework

(g) 8× super-resolution image with end-to-end framework

(h) 8× super-resolution map with end-to-end framework

Figure 4.6: Example results for the Vaihingen dataset with 8× up-scaling factor.



Road	0.86	0.015	0.041	0.035	0.037	0.0092
Tree	0.015	0.81	0.013	0.042	0.00029	0.11
Red roof	0.0023	0.00089	0.93	0.053	0.0035	0.0093
Grey roof	0.029	0.0033	0.054	0.83	0.0065	0.077
Concrete roof	0.014	0.0031	0.031	0.2	0.72	0.036
Vegetation	0.063	0.046	0.074	0.056	0.0053	0.75

(a) High-resolution data

Road	0.6	0.016	0.12	0.036	0.17	0.06
Tree	0.0026	0.0091	0.0033	0.00011	0.00025	0.98
Red roof	0.0047	0.0014	0.84	0.031	0.025	0.1
Grey roof	0.041	0.0019	0.08	0.096	0.18	0.6
Concrete roof	0.0013	3.4e-05	0.0032	0.00054	0.93	0.061
Vegetation	0.0015	0.00076	0.0031	0.00026	0.0024	0.99

(b) 4× interpolation

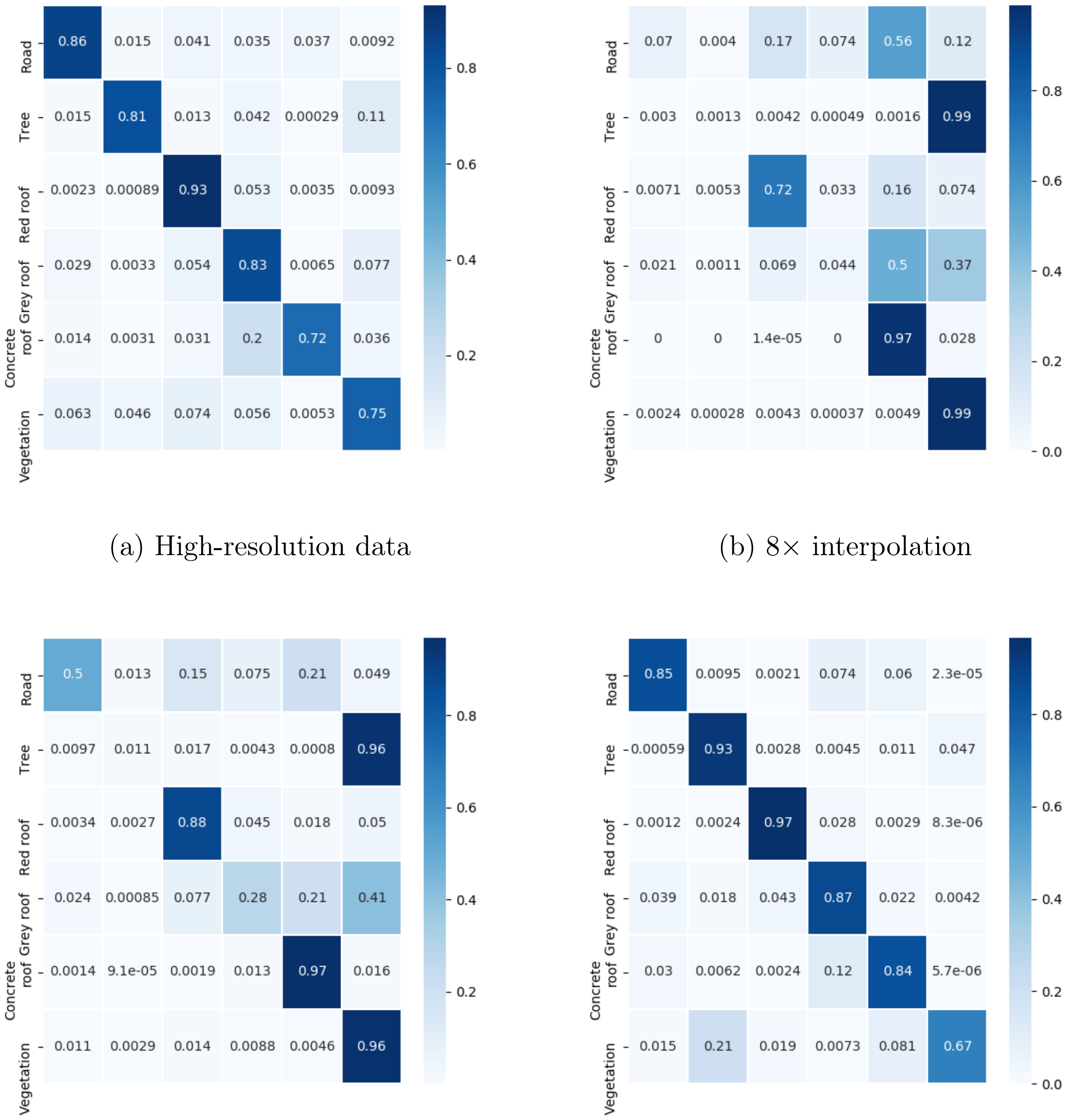
Road	0.77	0.01	0.063	0.026	0.089	0.041
Tree	0.0091	0.056	0.013	0.0041	9.8e-05	0.92
Red roof	0.0075	0.0023	0.85	0.065	0.013	0.064
Grey roof	0.026	0.0014	0.073	0.47	0.063	0.36
Concrete roof	0.0032	0.00025	0.0068	0.057	0.89	0.043
Vegetation	0.011	0.0045	0.014	0.009	0.0025	0.96

(c) 4× super-resolution with two stage framework

Road	0.91	0.0052	0.0017	0.027	0.06	0.00016
Tree	0.00051	0.91	0.0063	0.0025	0.0043	0.072
Red roof	0.013	0.0013	0.96	0.021	0.0085	2.3e-05
Grey roof	0.18	0.017	0.05	0.7	0.049	0.00013
Concrete roof	0.1	0.0013	0.0054	0.043	0.85	2e-05
Vegetation	0.03	0.15	0.025	0.0063	0.069	0.72

(d) 4× super-resolution with end-to-end framework

Figure 4.7: Accuracy heatmaps for semantic segmentation on the Thetford dataset with 4× up-scaling.



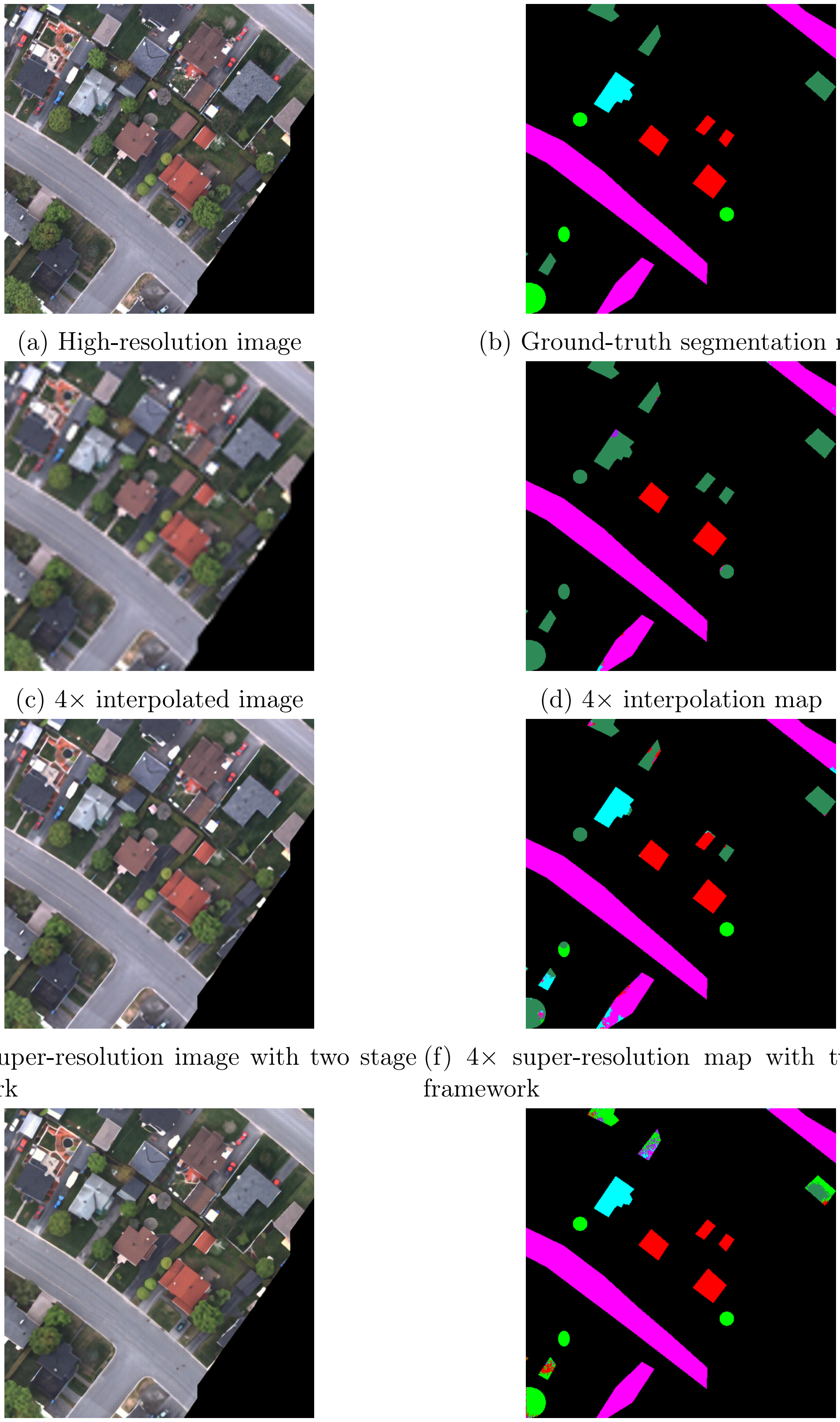
(a) High-resolution data

(b) 8× interpolation

(c) 8× super-resolution with two stage framework

(d) 8× super-resolution with end-to-end framework

Figure 4.8: Accuracy heatmaps for semantic segmentation on the Thetford dataset with 8× up-scaling.



(a) High-resolution image

(b) Ground-truth segmentation map

(c) 4× interpolated image

(d) 4× interpolation map

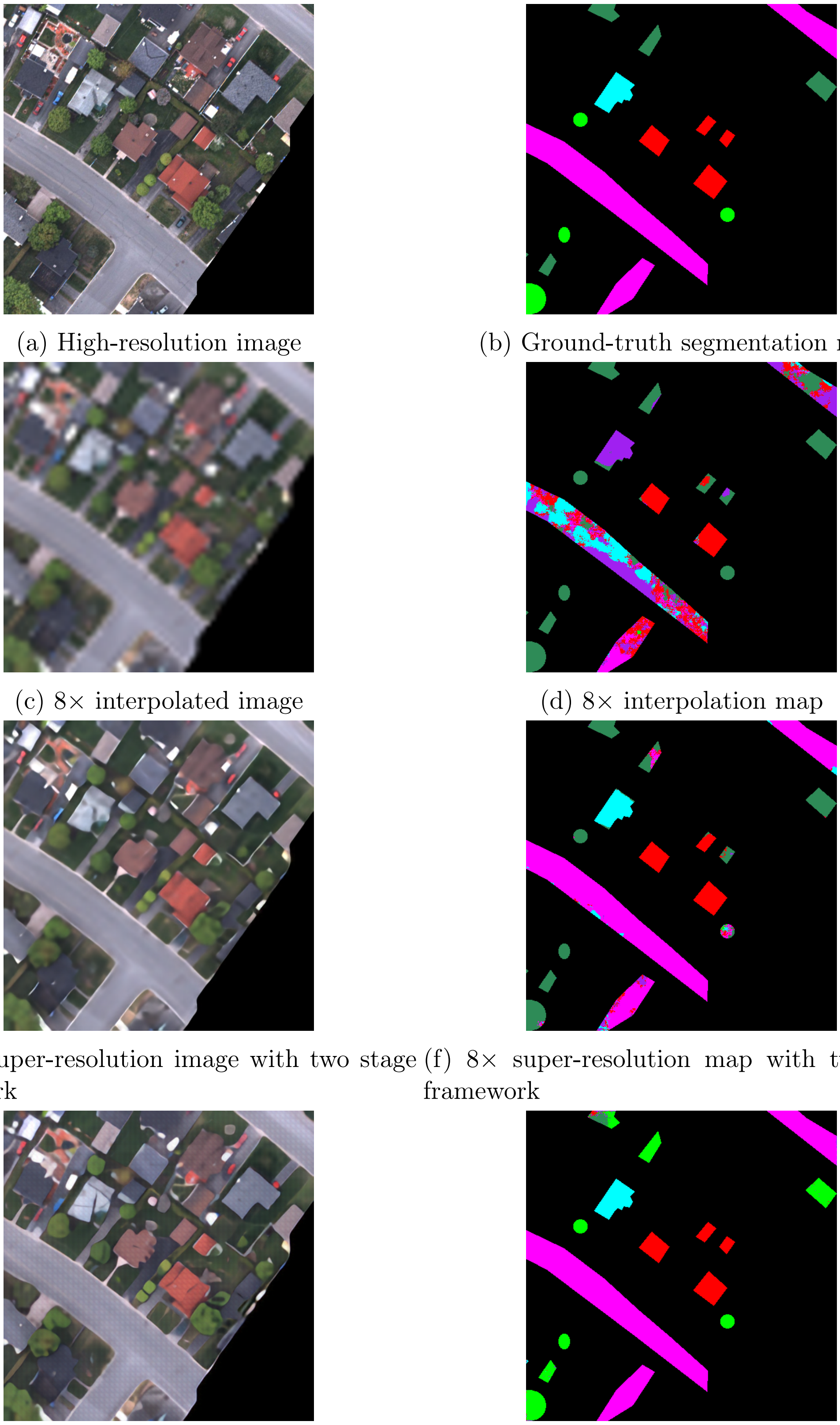
(e) 4× super-resolution image with two stage framework

(f) 4× super-resolution map with two stage framework

(g) 4× super-resolution image with end-to-end framework

(h) 4× super-resolution map with end-to-end framework

Figure 4.9: Example results for the Thetford dataset with 4× up-scaling factor.



(a) High-resolution image

(b) Ground-truth segmentation map

(c) 8× interpolated image

(d) 8× interpolation map

(e) 8× super-resolution image with two stage framework

(f) 8× super-resolution map with two stage framework

(g) 8× super-resolution image with end-to-end framework

(h) 8× super-resolution map with end-to-end framework

Figure 4.10: Example results for the Thetford dataset with 8× up-scaling factor.

Chapter 5

Conclusion and Future Works

In this work, we presented two frameworks that generate more accurate semantic segmentation thematic maps for low-resolution remote sensing inputs with the use of super-resolution. The first one uses super-resolution as a pre-processing step for the semantic segmentation task. The second one trains one single network that shares the loss of both super-resolution and semantic segmentation. We evaluated their performances on three highly different aerial datasets and under two degradation factors, comparing the results with low-resolution bicubic up-sampled inputs.

Super-resolution was confirmed to be a viable strategy to recover important texture and object details for semantic segmentation. With enough training data, the recovered texture information greatly helps the semantic segmentation not to mislabel similar classes. Small objects, such as the cars in the Vaihingen dataset, which are not easily detected on low-resolution representations, can become more discernible with the employment of super-resolution.

For either quantitative or qualitative evaluation, super-resolved inputs surpassed the low-resolution bicubic up-sampled ones in all cases. This improvement was more significant on 8× down-sampling factors, since the amount of information loss in the low-resolution representation is vast enough to negatively affect the generation of thematic maps, but also small enough to allow the reconstruction by the super-resolution network.

The two stage framework reconstructs the image based solely on the super-resolution loss. This means that the reconstruction aims to maximize the PSNR value, which usually implies a better image for human perception. However, as the end-to-end framework allows the semantic segmentation loss to also coordinate the image reconstruction, the super-resolved image for this framework presents artifacts generated by the semantic segmentation network (such as textures) that help the task to be performed. In these cases, the PSNR may even be worse than the one for a low-resolution image, but without compromising the segmentation performance.

Although the end-to-end framework seems to be better than the two stage one, it is important to remark that they evaluate slightly different situations. The semantic segmentation network of the two stage framework is trained with high-resolution data and is applied to low-resolution images, while the end-to-end is trained and tested with low-resolution data as input. Even so, the end-to-end approach performed better. Thus, this is an indication that in a real-world case in which we need to input low-resolution images to a network trained with high-resolution, it may be better to down-sample the available high-resolution data and perform the training again with the end-to-end pipeline.

For future work, there are more experiments that can be performed. In this work, we studied the case when a semantic segmentation network is trained with high-resolution data and needs to be applied to low-resolution images later (as in the evaluation of the two stage framework). Therefore, we plan to evaluate different cases, such as the performance of a semantic segmentation network when trained and tested with low-resolution data compared to the use of super-resolution. We can also evaluate the performance of a semantic segmentation network trained with super-resolved data and compare the results with the same network trained with low-resolution data only.

We also plan to apply the proposed frameworks in real-world data, instead of manually down-sampled images. Furthermore, we plan to use the other available bands during the training in order to improve the results even more.

Finally, there is also space to study how different reconstruction/visual benchmarks can help to enhance the semantic segmentation performance more than PSNR. Adversarial losses from GANs, for example, are an example of such benchmark, but that may create undesirable features in the super-resolved image.

Bibliography

Ahn, N., Kang, B., and Sohn, K.-A. (2018a). Fast, accurate, and lightweight super-resolution with cascading residual network. In *The European Conference on Computer Vision (ECCV)*.

Ahn, N., Kang, B., and Sohn, K.-A. (2018b). Image super-resolution via progressive cascading residual network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Audebert, N., Le Saux, B., and Lefèvre, S. (2017). Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In *Computer Vision – ACCV 2016*, pages 180--196. Springer International Publishing.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12).

Bosch, M., Gifford, C. M., and Rodriguez, P. A. (2018). Super-resolution for overhead imagery using densenets and adversarial learning. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1414–1422.

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018a). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *The European Conference on Computer Vision (ECCV)*.

Chen, R., Qu, Y., Zeng, K., Guo, J., Li, C., and Xie, Y. (2018b). Persistent memory residual network for single image super resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dai, D., Wang, Y., Chen, Y., and Van Gool, L. (2016). Is image super-resolution helpful for other vision tasks? In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9.

Dong, C., Loy, C. C., He, K., and Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307.

Feng, R., Gu, J., Qiao, Y., and Dong, C. (2019). Suppressing model overfitting for image super-resolution networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Ferdous, S. N., Mostofa, M., and Nasrabadi, N. M. (2019). Super resolution-assisted deep aerial vehicle detection . In Pham, T., editor, *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, pages 432 -- 443. International Society for Optics and Photonics, SPIE.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672--2680. Curran Associates, Inc.

Gu, J., Xu, G., Zhang, Y., Sun, X., Wen, R., and Wang, L. (2018). Wider channel attention network for remote sensing image super-resolution. *arXiv preprint arXiv:1812.05329*.

Han, W., Chang, S., Liu, D., Yu, M., Witbrock, M., and Huang, T. S. (2018). Image super-resolution via dual-state recurrent networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Haris, M., Shakhnarovich, G., and Ukita, N. (2018a). Deep backprojection networks for super-resolution. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Haris, M., Shakhnarovich, G., and Ukita, N. (2018b). Task-driven super resolution: Object detection in low-resolution images. *arXiv preprint arXiv:1803.11316*.

Haut, J. M., Fernandez-Beltran, R., Paoletti, M. E., Plaza, J., and Plaza, A. (2019). Remote sensing image superresolution using deep residual channel attention. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–13.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, J.-B., Singh, A., and Ahuja, N. (2015). Single image super-resolution from transformed self-exemplars. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hui, Z., Wang, X., and Gao, X. (2018). Fast and accurate single image super-resolution via information distillation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 723--731.

Jang, D.-W. and Park, R.-H. (2019). Densenet with deep residual channel-attention blocks for single image super resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Jiang, K., Wang, Z., Yi, P., Jiang, J., Xiao, J., and Yao, Y. (2018). Deep distillation recursive network for remote sensing imagery super-resolution. *Remote Sensing*, 10(11).

Kim, J., Kwon Lee, J., and Mu Lee, K. (2016a). Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646--1654.

Kim, J., Kwon Lee, J., and Mu Lee, K. (2016b). Deeply-recursive convolutional network for image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kim, J.-H. and Lee, J.-S. (2018). Deep residual network with enhanced upscaling module for super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, volume 7.

Kwak, J. and Son, D. (2019). Fractal residual network and solutions for real super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Lai, W.-S., Huang, J.-B., Ahuja, N., and Yang, M.-H. (2017). Deep laplacian pyramid networks for fast and accurate super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lanaras, C., Bioucas-Dias, J., Galliani, S., Baltsavias, E., and Schindler, K. (2018). Super-resolution of sentinel-2 images: Learning a globally applicable deep neural network. *arXiv preprint arXiv:1803.04271*.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A. P., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, volume 2, page 4.

Lei, S., Shi, Z., and Zou, Z. (2017). Super-resolution for remote sensing images via local–global combined network. *IEEE Geoscience and Remote Sensing Letters*, 14(8):1243–1247.

Lim, B., Son, S., Kim, H., Nah, S., and Lee, K. M. (2017). Enhanced deep residual networks for single image super-resolution. In *The IEEE conference on computer vision and pattern recognition (CVPR) workshops*, volume 1, page 4.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision – ECCV 2016*, pages 21--37. Springer International Publishing.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2017). High-resolution aerial image labeling with convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12):7092–7103.

Marmanis, D., Schindler, K., Wegner, J., Galliani, S., Datcu, M., and Stilla, U. (2018). Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:158 – 172.

Nogueira, K., Dalla Mura, M., Chanussot, J., Schwartz, W. R., and dos Santos, J. A. (2019). Dynamic multicontext segmentation of remote sensing images based on convolutional networks. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–18.

Pandey, G. and Ghanekar, U. (2018). A compendious study of super-resolution techniques by single image. *Optik*, 166:147 – 160. ISSN 0030-4026.

Park, D., Kim, K., and Chun, S. Y. (2018). Efficient module based single image super resolution for multiple problems. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, volume 5.

Penatti, O. A., Nogueira, K., and Dos Santos, J. A. (2015). Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 44--51.

Pouliot, D., Latifovic, R., Pasher, J., and Duffe, J. (2018a). Landsat super-resolution enhancement using convolution neural networks and sentinel-2 for training. *Remote Sensing*, 10(3).

Pouliot, D., Latifovic, R., Pasher, J., and Duffe, J. (2018b). Landsat super-resolution enhancement using convolution neural networks and sentinel-2 for training. *Remote Sensing*, 10(3):394.

Richards, J. A. and Jia, X. (1999). *Remote sensing digital image analysis*, volume 3. Springer.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234--241. Springer International Publishing.

Schowengerdt, R. A. (2006). *Remote sensing: models and methods for image processing.* Elsevier.

Seif, G. and Androutsos, D. (2018a). Large receptive field networks for high-scale image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 763--772.

Seif, G. and Androutsos, D. (2018b). Large receptive field networks for high-scale image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Sharma, M., Mukhopadhyay, R., Upadhyay, A., Koundinya, S., Shukla, A., Chaudhury, S., and CSIR-CEERI, P. (2018). Irgun: Improved residue based gradual up-scaling network for single image super resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, volume 8.

Shermeyer, J. and Van Etten, A. (2018). The effects of super-resolution on object detection performance in satellite imagery. *arXiv preprint arXiv:1812.04098*.

Sherrah, J. (2016). Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery. *arXiv preprint arXiv:1606.02585*.

Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shocher, A., Cohen, N., and Irani, M. (2018). Zero-shot" super-resolution using deep internal learning. In *Conference on computer vision and pattern recognition (CVPR)*.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Tai, Y., Yang, J., and Liu, X. (2017). Image super-resolution via deep recursive residual network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Timofte, R., Agustsson, E., Gool, L. V., Yang, M., Zhang, L., Lim, B., Son, S., Kim, H., Nah, S., Lee, K. M., Wang, X., Tian, Y., Yu, K., Zhang, Y., Wu, S., Dong, C., Lin, L., Qiao, Y., Loy, C. C., Bae, W., Yoo, J., Han, Y., Ye, J. C., Choi, J., Kim, M., Fan, Y., Yu, J., Han, W., Liu, D., Yu, H., Wang, Z., Shi, H., Wang, X., Huang, T. S., Chen, Y., Zhang, K., Zuo, W., Tang, Z., Luo, L., Li, S., Fu, M., Cao, L., Heng, W., Bui, G., Le, T., Duan, Y., Tao, D., Wang, R., Lin, X., Pang, J., Xu, J., Zhao, Y., Xu, X., Pan, J., Sun, D., Zhang, Y., Song, X., Dai, Y., Qin, X., Huynh, X., Guo, T., Mousavi, H. S., Vu, T. H., Monga, V., Cruz, C., Egiazarian, K., Katkovnik, V., Mehta, R., Jain, A. K., Agarwalla, A., Praveen, C. V. S., Zhou, R., Wen, H., Zhu, C., Xia, Z., Wang, Z., and Guo, Q. (2017). Ntire 2017 challenge on single image super-resolution: Methods and results. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1110–1121.

Tong, T., Li, G., Liu, X., and Gao, Q. (2017). Image super-resolution using dense skip connections. In *The IEEE International Conference on Computer Vision (ICCV)*.

Wang, X., Gu, Y., Gao, X., and Hui, Z. (2019a). Dual residual attention module network for single image super resolution. *Neurocomputing*, 364:269--279.

Wang, X., Yu, K., Dong, C., and Loy, C. C. (2018a). Recovering realistic texture in image super-resolution by deep spatial feature transform. *arXiv preprint arXiv:1804.02815*.

Wang, Y., Perazzi, F., McWilliams, B., Sorkine-Hornung, A., Sorkine-Hornung, O., and Schroers, C. (2018b). A fully progressive approach to single-image super-resolution. *arXiv preprint arXiv:1804.02900*.

Wang, Z., Chen, J., and Hoi, S. C. (2019b). Deep learning for image super-resolution: A survey. *arXiv preprint arXiv:1902.06068*.

Wen, R., Fu, K., Sun, H., Sun, X., and Wang, L. (2018). Image superresolution using densely connected residual networks. *IEEE Signal Processing Letters*, 25(10):1565–1569.

Xiao, A., Wang, Z., Wang, L., and Ren, Y. (2018). Super-resolution for "jilin-1" satellite video imagery via a convolutional network. *Sensors*, 18(4):1194.

Xu, X. and Li, X. (2019). Scan: Spatial color attention networks for real single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Yang, J., Wright, J., Huang, T. S., and Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873.

Yu, J., Fan, Y., Yang, J., Xu, N., Wang, Z., Wang, X., and Huang, T. (2018). Wide activation for efficient and accurate image super-resolution. *arXiv preprint arXiv:1808.08718*.

Yuan, Y., Liu, S., Zhang, J., Zhang, Y., Dong, C., and Lin, L. (2018). Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 701–710.

Zeyde, R., Elad, M., and Protter, M. (2012). On single image scale-up using sparse-representations. In Boissonnat, J.-D., Chenin, P., Cohen, A., Gout, C., Lyche, T., Mazure, M.-L., and Schumaker, L., editors, *Curves and Surfaces*, pages 711--730. Springer Berlin Heidelberg.

Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., and Fu, Y. (2018a). Image super-resolution using very deep residual channel attention networks. In *The European Conference on Computer Vision (ECCV).*

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. (2018b). Residual dense network for image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Zhang, Z., Wang, X., and Jung, C. (2019). Dcsr: Dilated convolutions for single image super-resolution. *IEEE Transactions on Image Processing*, 28(4):1625–1635.